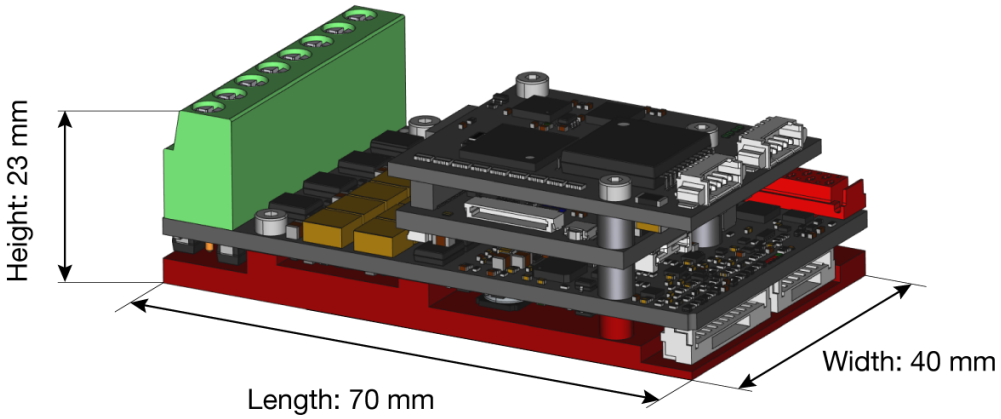
Technical Specifications



Power Specifications

Node Size	Node 400	Node 1000	Node 2000
Maximum Peak Output Power	415 W	1,040 W	2,080 W
Maximum Continuous Output Power *	415 W	1,040 W	1,200 W
Nominal Supply Voltage **	12-48 V DC		
Maximum Supply Voltage **	55 V DC cont.; 60 V DC peak		
Minimum Supply Voltage **	10 V DC		
Maximum Input Current DC	9.6 A DC	24 A DC	48 A DC
Maximum Phase Current RMS	13.2 A RMS	33 A RMS	66 A RMS ***
Maximum Continuous Phase Current RMS *	13.2 A RMS	33 A RMS	36 A RMS
Maximum Efficiency	> 98 %		

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

* The maximum continuous values that can be reached depend on the thermal integration . Please review the thermal mounting guidelines for details.

** Review these supply voltage definitions.

*** Refer to the load cycle.

General specifications

Supported Communication Standard	EtherCAT (EtherCAT is a registered trademark and patented technology licensed by Beckhoff Automation GmbH, Germany.)	
Number of Phases	4 1	
Supported Motors	1 x 3-phase BLDC-Motor and 1 active brake	
Brake control voltage	0-48 V PWM via Phase D, up to 10 A	
Analog Inputs	4 (2 x single-ended 0-10V, 2 x differential ±5V) 2	
Digital IOs	4 × GPIO (3.3 V CMOS logic) 3	
Encoder Connectors	2 x 4ch. RS-422	
Supported Standard Encoder Interfaces	Incremental Encoder (ABI), HALL, BiSS-C, SSI	
Supported Encoder Types	**Encoder Port 1** Incremental Encoder (ABI), BiSS-C, SSI, A-Format	**Encoder Port 2** Incremental Encoder (ABI), HALL, BiSS-C, SSI
Incremental Encoder maximum frequency	1.4 MHz 4 (supports up to 14,000 rpm with a 4,096 resolution incremental encoder)	
Hardware Protections	Overcurrent, Overvoltage, Undervoltage, Temperature	
Compliance with standards:	CISPR 11 Class B (EN 55011:2016) IEC 61000-4-6:2013 IEC 61000-4-3:2020 IEC 61000-4-2:2008	

	IEC 61000-4-8:2009 IEC 61800-5-1:2007 IEC 60204-1:2016
Compliance with European directives:	CE (EMC Directive 2014/30/EU)
Certificates	UL listed (Node 400, Node 1000 with soldered cables, Node 2000 with soldered cables) UL recognized (Node 1000, Node 2000)
Dimensions	70 x 40 x 23 mm
Weight	85 g (incl. Heatsink)
Operating temperature	0 °C 5 to 40 °C
Storage Temperature	-35 to 85 °C
Maximum installation altitude	2000 m
Humidity	5-85% rH
Sensors on-board	DC-BUS Voltage, DC-BUS current, Phase Voltage, Phase current (two phases measured, one computed), Temperature

1 The fourth phase can be used for a brake or other purposes (see: Manually controlling phase D voltage); it has a maximum current of 10 A.

2 All Analog Inputs can be configured as single-ended 0-5 V, 0-10 V, 0-20 V or differential ±5 V, ±10 V independently. Upon request only, please contact support@synapticon.com.

3 Two Digital IO can be configured as 5.0 V CMOS logic. Upon request only, please contact support@synapticon.com.

4 Depends on the firmware used. Value given for v4.1

[5]

The actual lowest temperature is limited by the dew point. This depends on several factors, e.g. the ambient humidity and how quick the drive cools down after operation. Special care must be taken when the servo drive is actively cooled below ambient temperature.

Load cycle

The maximum peak current is achievable for at least 10 seconds. The maximum allowable **duty cycle** depends on the cooling situation and load cycle profile. Under standard conditions (150 mm * 200 mm * 20 mm aluminium plate, 20° C ambient temperature, square wave load cycle alternating between 0 and max current), a duty cycle of 10% is achievable. The value can be higher for optimized system setups, e.g. 25% when using a SOMANET Node variant with **directly soldered power wires** or up to 100% if **the mounting plate is actively cooled additionally**. Please contact support@synapticon.com for details.

Downloads

PDF

Here you can download the entire documentation as PDF:

DOWNLOAD PDF SOMANET NODE

Documents

DECLARATION OF CONFORMITY SOMANET NODE SAFETY

CERTIFICATE SOMANET NODE SAFETY

EC-TYPE EXAMINATION

STATEMENT SOMANET NODE MECHANICAL IMMUNITY (SHOCK AND VIBRATION TESTS)

CAD Files

STEP MODEL SOMANET NODE

STEP MODEL SOMANET NODE SAFETY

UL

CERTIFICATE OF COMPLIANCE UL-LISTED SOMANET NODE

CERTIFICATE OF COMPLIANCE UL-RECOGNIZED SOMANET NODE

Hardware Installation Manual

Attention:

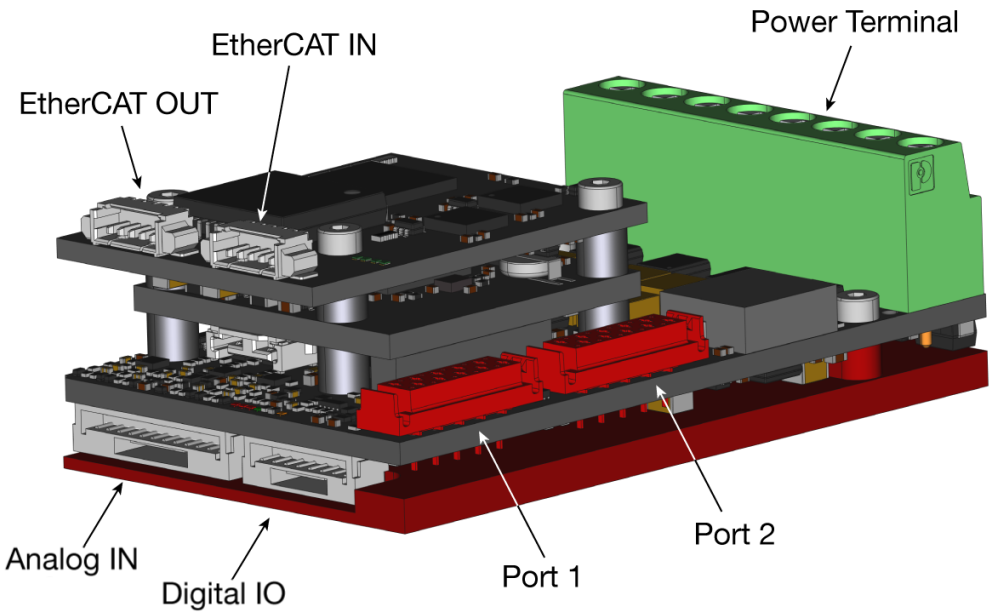
When dealing with electronic devices, always consider these handling instructions!

Wiring instructions 6
Overview of connectors 6
Power and brake 6
Encoders and IO 11
Mating parts 20
Mounting instructions 21
Heat dissipation 21
Dimensions 21
Interference with magnetic fields 23
Thermal mounting considerations 24
On-site installation guide (UL) 25
Branch fuse required 26
Field wiring terminal marking 27
LED signals 28
Overview 28
Core LED 29
Drive LED 30
Com LED 31
Hardware Diagrams 31
EtherCAT module 32
Processor module 33
Drive module 34
SOMANET Node Safety (Original instructions) 35

Dimensions 36
Specifications of the safety functions 37
Setup of the safety functions 41
Using the safety functions 46
Examples for realising safety functions 51
Commissioning and maintenance 54
Changelog of the safety-related documentation 54

Wiring instructions

Overview of connectors

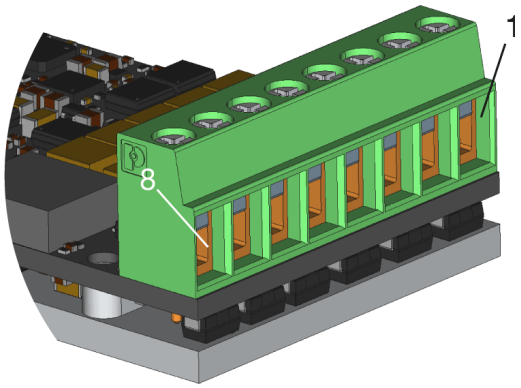


- Power and brake
- Encoders and IO
- Connector types and mating parts numbers

Power and brake

Pinout Power Terminal

> Important:
>
> The cable length for Main Power Supply (12-48 V) must not exceed 30 m.



Important:

The power cables must be twisted for EMC reasons.

Attention:

No reverse polarity protection! Always ensure that the power cables are connected correctly, otherwise the Drive may be damaged beyond repair!

Terminal #	
1	Phase A
2	Phase B
3	Phase C
4	Phase D - Brake output 0-48 V PWM
5	Ground
6	Main Power Supply 12-48 V
7	Optional Logic Ground *
8	Optional Logic Supply 12-24 V **

* Maximum current carrying capacity: 10 A

Note:

** this additional logic supply is not connected by default. It is not required for normal operation. For additional information, please contact sales@synapticon.com

Note:

SOMANET servo drives are densely packed with high-performance components. Even at idle/torque off, several components such as power supplies, cause the servo drive to get warm, especially when it's not connected to any heat conducting structure. The servo drives' power stages and control algorithms were optimized for efficiency at high power output and actually make them one of the most efficient servo drives available. The heat generation is not linear, which makes it actually more noticeable at idle or low power. Please ensure by design of your mounting situation that the heat interface is connected to sufficient structure to appropriately dissipate generated heat.

Notes about choosing the supply voltage

The indicated minimal supply voltage of 10 V is enough to power on the servo drive for configuration but **for turning a motor, a voltage above 12 V is required**.

Note:

Due to voltage drop from power supply to the servo drive is it not recommended to supply the servo drive at absolute minimal voltages. If your system does not run at 12 V, increasing this value slightly may be necessary.

Using a PELV or SELV power supply

It's recommended to use a protected extra-low voltage supply (PELV) instead of a safety extra-low voltage supply (SELV).

If a SELV is used, it's possible that the isolation from the earth can be violated through our drive (e.g. through heatsink or mechanical mountings) and the supply becomes PELV.

Warning about using contactors behind the power supply

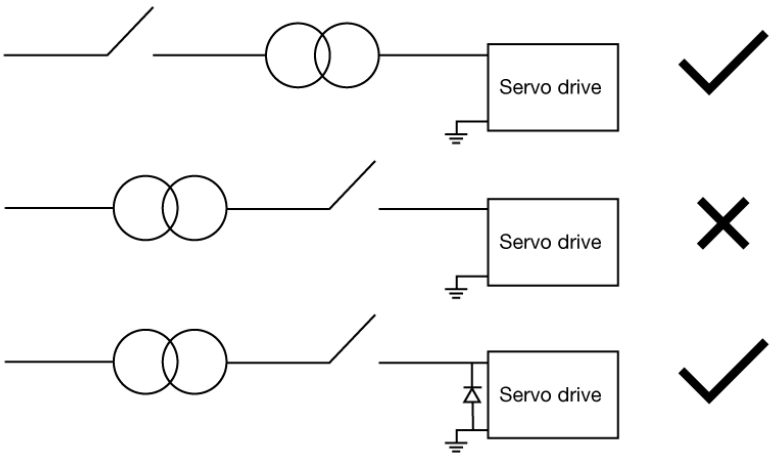
Attention

Attention:

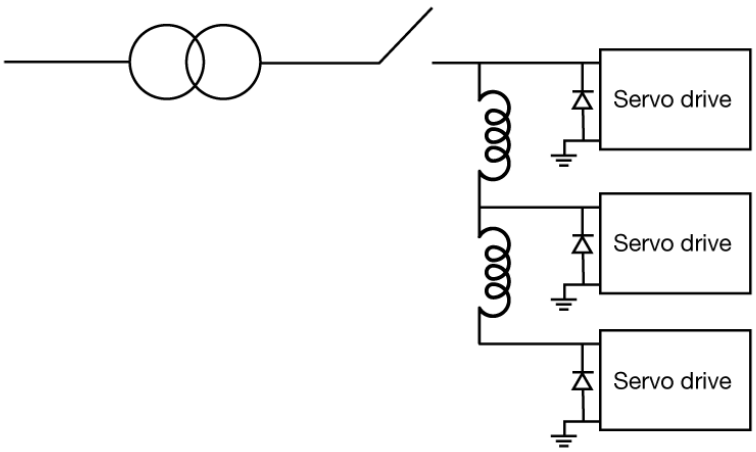
Our servo drives are designed for voltages between 24 V_{DC} and 48 V_{DC} (55 V_{DC}; Max,-cont 60 V_{DC} Max,peak), they should be run with an appropriate extra-low voltage supply. Please **do not use contactors behind the power supply** since the transient-voltage-suppression diodes could get damaged due to the power-up voltage increase (Surge). This is likely to happen when the power up occurs fast and **can lead to complete failure** of your servo drive.

If contactors behind the power supply are used, it's necessary to **include an uni-directional TVS diode type 1.5KE62A-E3/54*** between Main Power Supply and Ground of the terminal.

* this model has been successfully tested by Synapticon. Other products with the same specifications may also be appropriate but can't be recommended.



In case you are running several servo drives behind a contactor, please use an uni-directional TVS diode on **each drive** because of the inductances of the wiring.

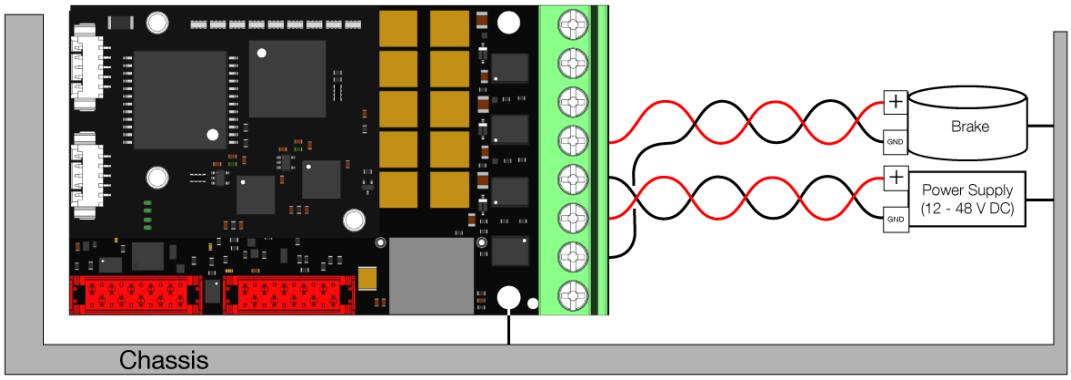


Connecting a Brake

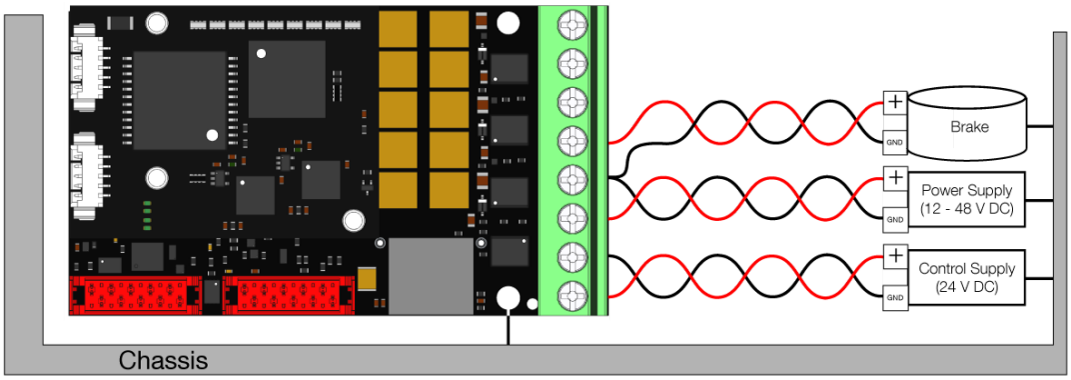
> Attention:
>
> When using a safe brake in combination with Node Safety, use SBC Out instead of Phase D.

If your system has an attached brake, please connect the brake cables to Phase D and Ground. These two threads should be twisted together or at least be paired to have a minimum area between them.

By default, Logic Supply is deactivated, it is therefore recommended to use Logic Ground (pin 7 of the Main Supply Connector) for the brake.



If Logic Supply is active, Ground of the brake can be connected to the Ground cable of the 48V power supply in a spot close to the board or it can be connected together with the Ground of the 48V supply to the pin 5 of the Main Supply connector if they fit in there.

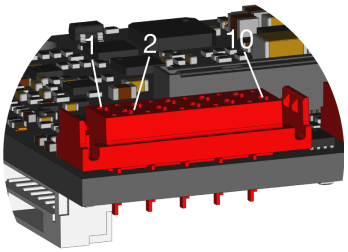


Important:

Please make sure the servo drive is properly grounded. You can connect the heatsink to ground.

Encoders and IO

Encoder Port 1



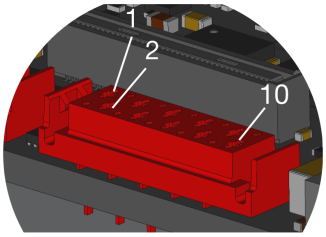
Pin #	ABI *	BiSS *	SSI *	Half-Duplex / A-Format *
1	•	MA-	Clock-	SD- / A
2	5 V **	5 V **	5 V **	5 V **
3	Ground	Ground	Ground	Ground
4	•	MA+	Clock+	SD+ / B
5	A-	SLO-	Data-	•
6	A+	SLO+	Data+	•

Pin #	ABI *	BiSS *	SSI *	Half-Duplex / A-Format *
7	B-	•	•	•
8	B+	•	•	•
9	I-	•	•	•
10	I+	•	•	•

* This is a differential line according to RS422/RS485 standards

** This is a 5 V supply that can provide up to 250 mA for external use. The supply is protected against short to ground and keeps the current below 400 mA in a continuous short.

Encoder Port 2

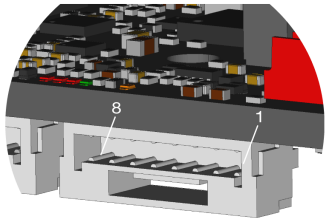


Pin #	HALL	ABI (dif-ferential)*	ABI (single-ended)**	BiSS *	SSI *
1	•	•	•	MA-	Clock-
2	5 V ***	5 V ***	5 V ***	5 V ***	5 V ***
3	Ground	Ground	Ground	Ground	Ground
4	•	•	•	MA+	Clock+
5	C/H3	A-	A	SLO-	Data-
6	•	A+	•	SLO+	Data+
7	B/H2	B-	B	•	•

Pin #	HALL	ABI (dif-ferential)*	ABI (single-ended)**	BiSS *	SSI *
8	•	B+	•	•	•
9	A/H1	I-	I	•	•
10	•	I+	•	•	•

* This is a differential line according to RS422/RS485 standards
** This is a single-ended TTL line with 1 kΩ pull-up resistors
*** This is a 5 V supply that can provide up to 250 mA for external use. The supply is protected against short to ground and keeps the current below 400 mA in a continuous short.

Encoder Port 3 / Digital IO



Pin #	GPIO	REM 14/16MT	BiSS (LVTTL)	SSI (LVTTL)	SPI	Type of Port
1	3.3 V *	3.3 V *	3.3 V *	3.3 V *	3.3 V *	•
2	Digital IO 1**	SS	SLO	Data	SS/CS	(Fast Push-Pull)
3	Digital IO 2**	SCK	MA	Clock	SCK	Fast port
4	Ground	Ground	Ground	Ground	Ground	•
5	Digital IO 3**	MISO	•	•	MISO	Fast port

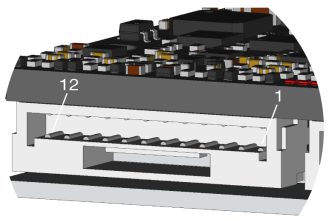
Pin #	GPIO	REM 14/16MT	BiSS (LVTTL)	SSI (LVTTL)	SPI	Type of Port
6	Digital IO 4**	MOSI	•	•	MOSI	Fast port
7	5 V ***	5 V ***	5 V ***	5 V ***	5 V ***	•
8	•	•	•	•	•	•

All Digital IOs are 3.3 V LVTTL CMOS logic and can be used for data rates up to 20 Mbps.
* This is a +3.3 V supply that can provide up to 250 mA for external use. The supply is protected against short to ground and keeps the current below 400 mA in a continuous short.
** Can be configured as 5.0 V CMOS logic. Upon request only, please contact sales@-synapticon.com
*** This is a +5 V supply that can provide up to 250 mA for external use. The supply is protected against short to ground and keeps the current below 400 mA in a continuous short.

> Note:
>
> Each Digital IO needs to be configured as a Digital Input or Output via object 0x2210 (“GPIO”). The object can also be used to activate a pull-down resistor.
>
> Changes may require to powercycle the servo drive.

Analog IN

> Note:
>
> Details about measuring Analog signals can be found in the system integration guideline: Connecting and configuring analog inputs



Pin #		Default Setting *	Object Mapping
1	•		
2	Analog Input 1 -	Connect to Ground **	
3	Analog Input 1 +	Single-ended 0-10 V	Analog input 1 (0x2401)
4	Ground		
5	Analog Input 2 -	Connect to Ground **	
6	Analog Input 2 +	Single-ended 0-10 V	Analog input 2 (0x2402)
7	5 V ***		
8	10 V ****		
9	Analog Input 3 -	Differential ±5 V	Analog input 3 (0x2403)
10	Analog Input 3 +	Differential ±5 V	Analog input 3 (0x2403)
11	Analog Input 4 -	Differential ±5 V	Analog input 4 (0x2404)
12	Analog Input 4 +	Differential ±5 V	Analog input 4 (0x2404)

* All Analog Inputs can be configured as single-ended 0-5 V, 0-10 V, 0-20 V or differential ±5 V, ±10 V independently. Upon request only, please contact sales@synapticon.com

** Must be connected to Ground when the Analog Input is set to Single-Ended Mode. It is recommended to use a separate ground cable for each single-ended analog signal/cable and twist or tie them together.

*** This is a 5 V supply that can provide up to 100 mA for external use. The supply is protected against short to ground and keeps the current below 400 mA in a continuous short.

**** This is a 10 V supply that can provide up to 25 mA for external use. The supply is protected against short to ground and keeps the current below 100 mA in a continuous short.

Analog input specification

Differential

Symbol	Min.	typical	Max.	Description	Impact
V_{diff}	-10 V		+10 V	The input voltage between the positive and negative input pins.	Any voltage out of this range causes saturation to min/max ADC value
V_{cm}	-10 V	0 V	+10 V	The common mode voltage.	Any voltage out of this range causes saturation and inaccuracy in the measurements.
$V_{in,P}$	-15 V		+15 V	The tolerable voltage between the positive input pin and GND.	Any voltage out of this range damages the analog input's circuitry.
$V_{in,N}$	-15 V		+15 V	The tolerable voltage between the negative input pin and GND.	Any voltage out of this range damages the analog input's circuitry.

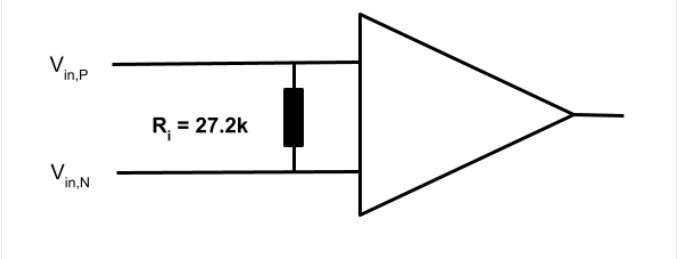
Symbol	Min.	typical	Max.	Description	Impact
R_i		27.2 kΩ		Internal input resistance	
Error	-1%		+1%	Error in ticks	

Single-ended

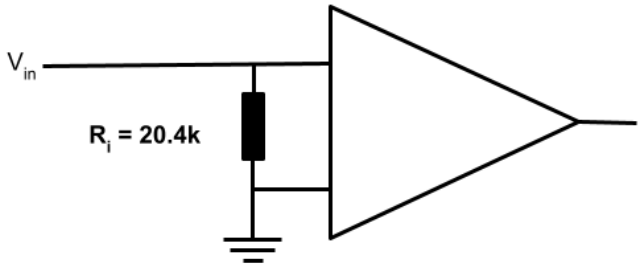
Symbol	Min.	typical	Max.	Description	Impact
V :sub`in`	0 V		+1 V	The input voltage between the input and GND.	Any voltage out of this range causes saturation to min/max ADC value
$V_{in,lim}$	-15 V		+15 V	The tolerable input voltage between the input and GND.	Any voltage out of this range damages the analog input's circuitry.
R_i		20.4 kΩ		Internal input resistance	
Error	-1%		+1%	Error in ticks	

Analog input resistance

Differential



Single-ended



How to connect a temperature sensor

There is no particular input for temperature sensors in Node. Four analog inputs are available. Any of these can be used to measure the temperature sensor resistance. But the single-ended (AI 1, AI 2) is more accurate than the differential inputs (AI 3, AI 4) because the voltage difference is less. A simple circuit with an extra resistor for voltage dividing has to be implemented **externally**. For details about using temperature sensors please check the System integration guide.

Attaching a voltage divider

A voltage divider is needed when the Analog In needs to be calibrated. This is required for adjusting the measuring range when using a temperature sensor.

> Attention:
>
> The ADC value is scaled to 16 bit.

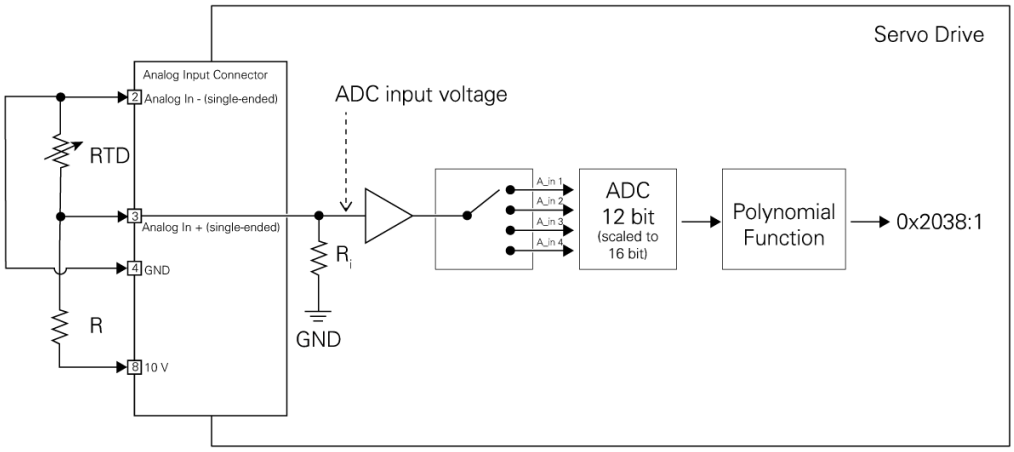
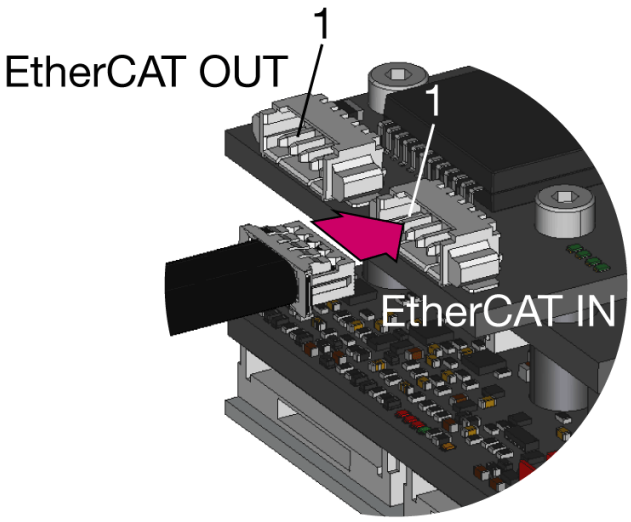


Image 1 - how to connect an RTD sensor resistance using the single-ended analog input

The internal resistance R_i of the single-ended analog input is 20.4 kΩ and it is connected in parallel to the RTD sensor.

EtherCAT port



EtherCAT IN Port

Pin	Name	Function
1 *	TX+	Differential transmit pair, positive line
2	TX-	Differential transmit pair, negative line
3	Shield	
4	RX+	Differential receive pair, positive line
5	RX-	Differential receive pair, negative line

* Pin 1 is on left

EtherCAT OUT Port

Pin	Name	Function
1 *	TX+	Differential transmit pair, positive line
2	TX-	Differential transmit pair, negative line
3	Shield	
4	RX+	Differential receive pair, positive line
5	RX-	Differential receive pair, negative line

* Pin 1 is on left

Mating parts

Connector details and mating part numbers

Connector description	Manufacturer	Mating part	Crimping Contact	Cable
Analog IN	JST Sales America Inc.	GHR-12V-S	SSHL-002T-P0.2	26-30 AWG *
Encoder Ports 1 & 2	TE Connectivity	1-215083-0	N/A	Flat Cable 28 AWG P1.27
Port 3 (Digital IO)	JST Sales America Inc.	GHR-08V-S	SSHL-002T-P0.2	26-30 AWG *
Power Ter-minal Block	Phoenix Contact	N/A		5859 WH005 (Alpha Wire)
EtherCAT IN/OUT	Molex	510210500 **	500588000	28-32 AWG

* Crimped cable from JST Sales America can be used, Parts number: AGHGH28K305

> Note:
>
> ** Housing only. Please ensure to only use shielded cables for EtherCAT communication!
>
> Please also check out our dedicated EtherCAT node to PC cable which is specifically designed for this purpose.

Mounting instructions

> Attention:
>
> Servo drives should not be mounted in direct sunlight.

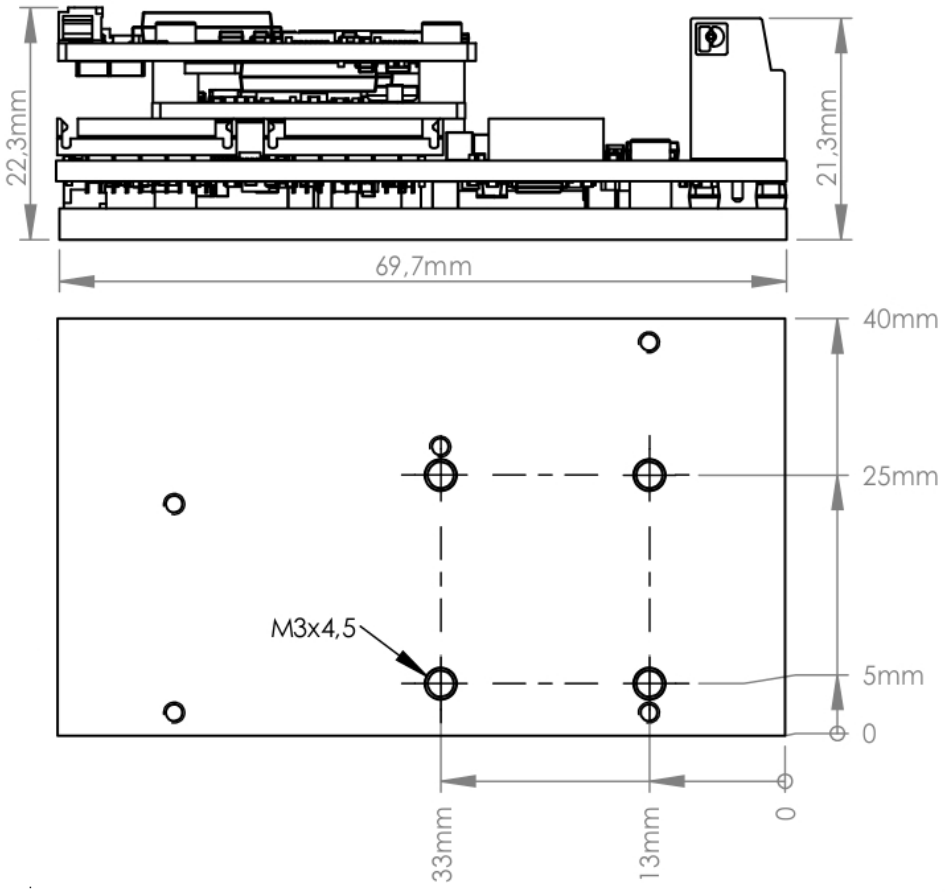
Heat dissipation

For correct functioning, the servo drive must be installed on a sufficiently dimensioned heat conducting surface made of a suitable material, such as aluminum.

Ensure that the servo drive is screwed properly to the heat conducting surface to facilitate the heat transfer.

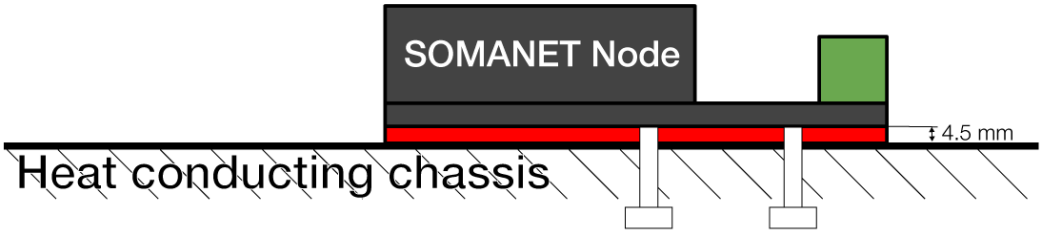
Dimensions

> Note:
>
> The safety version of Node has a different geometry which you can find here.



Note:

Please ensure to use as much of the available thread length as possible for tightening the Node to prevent damage to the thread.



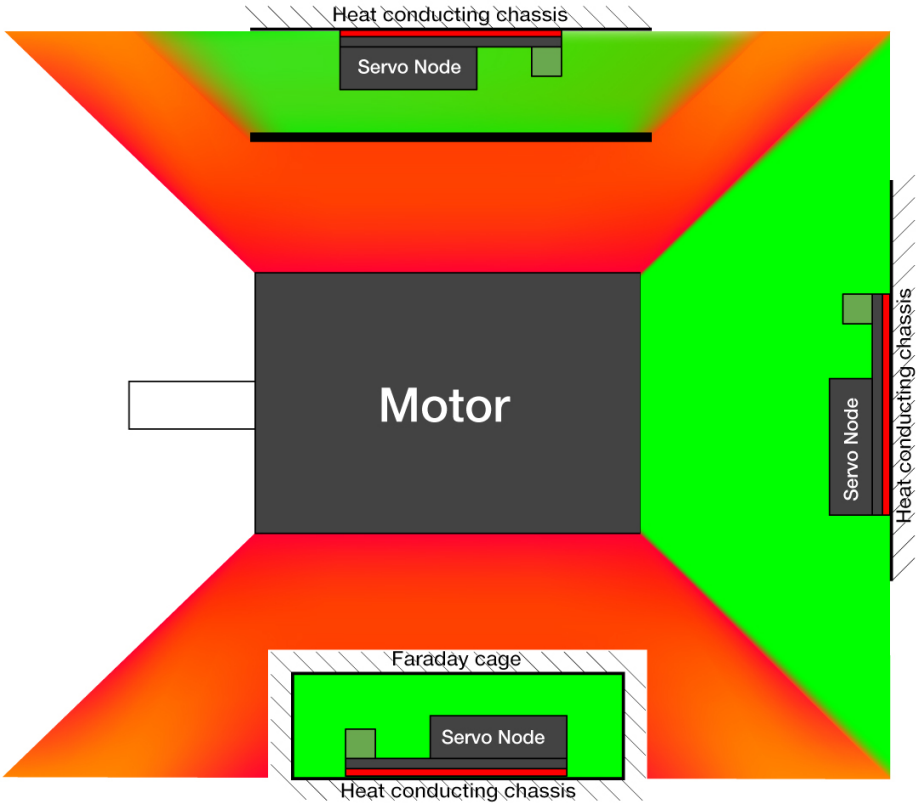
Interference with magnetic fields

Please consider the placement of the servo drive carefully when designing the motion control application. It has become evident that the motor's magnetic field has an impact on the servo drive's performance.

> Attention:
>
> Please also consider the magnetic fields that emanate from the brake.

These are the preferred arrangements:

1. At any position but shielded with a Faraday's cage
2. Behind the motor
3. Beside the motor but shielded from the motor's magnetic field with a plate consisting of a magnetic shielding material such as steel or Mu-Metal.



Thermal mounting considerations

While the maximum peak current is a fixed property of a servo drive, the maximum continuous output current is highly dependent on its thermal integration into the surrounding system. The boundary conditions described below are idealized scenarios that can be described and defined clearly. In most cases they will not represent the real-world application exactly, as the geometric and thermodynamic situation in a robot or other application is typically more complex. The purpose of these exemplary scenarios is to provide the system designer with an estimation of the required cooling capacity of the structure surrounding the SOMANET servo drive.

1. Application integration defined by interface temperature

   In this scenario, the servo drive is assumed to be integrated into a structure that is cooled externally to a defined maximum temperature. The defined continuous currents are achievable as long as the drive is bolted to a metal structure using all 4 M3 screws, the entire red aluminum bottom is in contact with the structure and all of the contact surface cooled to no hotter than the temperature specified in the figure below. This way of specification is the most accurate one and can directly be used in the design process of the system.

2. Application integration on a sample plate in air *

   2.a Application Integration Defined by Ambient Temperature

   In this scenario, the servo drive is mounted to a geometrically defined structure. As “real-world structures” are typically too complex to reflect them here, a simple geometry (rectangular aluminum plate of 250 x 250 x 20 mm) was selected. The drive is bolted to the center of a metal plate of such geometry using all 4 M3 screws, the plate is mounted vertical and the rest of the metal plate is surrounded by ambient air. Depending on the temperature of the ambient air, the continuous currents specified in the figure are achievable. It will be subject to free convection cooling. The available continuous currents can be further increased when adding active cooling (fan).

   2.b Application Integration Defined by Geometry

   As in the scenario above, the drive is mounted to the center of a vertical aluminum plate. Here, a fixed ambient temperature (20°c) is assumed, but different sizes of the plate are considered. Depending on the size of the plate, the continuous currents specified in the figure are achievable. The available continuous currents can be further increased when adding active cooling (fan).

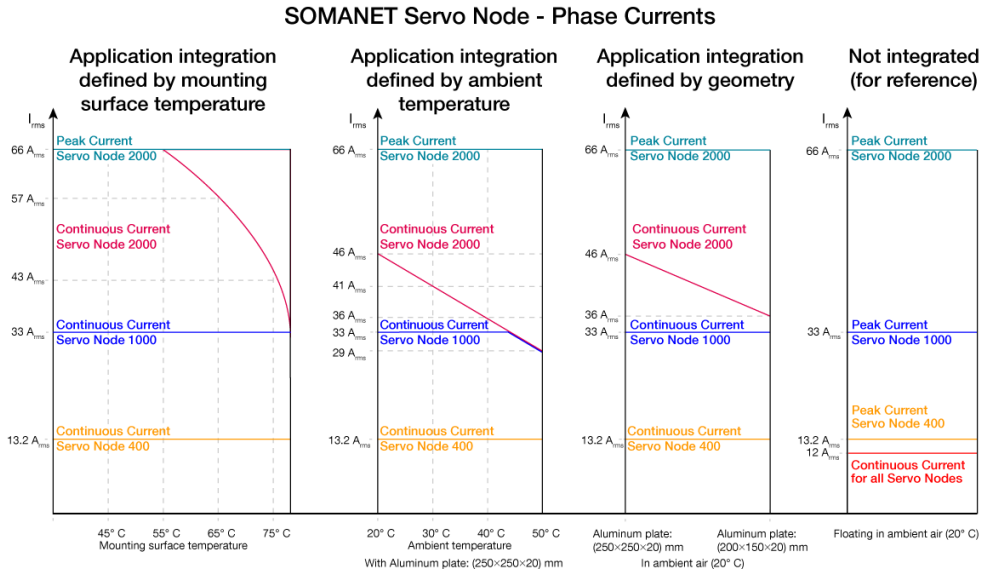
3. Servo drive floating in ambient air

This is a reference value when not integrating the node into any structure. As SOMANET servo drives are designed as embedded products, this situation is not recommended for both thermal and mechanical reasons!

Attention

A similarly bad cooling situation will be created if the node is mounted to a structure with poor thermal conductivity, in particular **structures made of plastic.**

* The continuous phase current given in the specsheet provided by Synapticon assumes this kind of boundary condition with a 200 x 150 x 20 mm aluminum mounting plate.



On-site installation guide (UL)

Please regard this information when installing the Node.

Installation specifications	Node 400	Node 1000	Node 2000
E-File number	E506869		
Rating in Volt	12-48 V		
Rating in Ampere	13.2 A_{rms}	33 A_{rms}	66 A_{rms}
Short circuit current rating	5 kA		
Maximum Surrounding Air Temperature	40° C		

Installation specifications	Node 400	Node 1000	Node 2000
Torque Values for Terminal screws	0.5 - 0.6 Nm (4.4 lbf·in)		
Short circuit rating and fuse type	Listed Class T Fuse, rated min. 160 Vdc 50 A 20 kA. (JDDZ/7.E81895)		
Temperature rating of field installed conductors >85°	N/A		

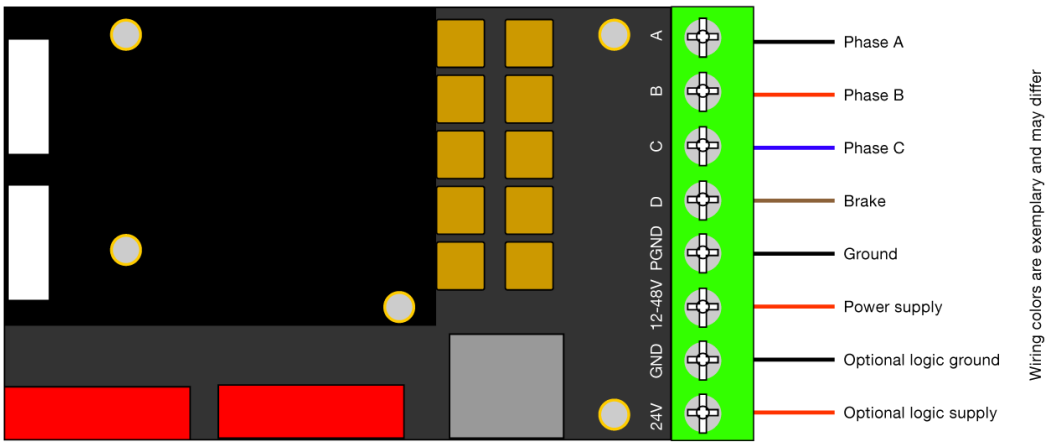
Branch fuse required

SOMANET servo drives don't have a fuse on-board, so a branch fuse is required as a protection.

The following fuse has been tested and approved by UL according to the requirements of UL 61800-5-1:

Manufacturer	Littelfuse
Product designation	JLLN050.T
CCN	JDDZ/7
Class	T
Cat. No.	JLLN, 558
V_{AC}	300 V
V_{DC}	160 V
Interrupt Rating (IR)	20 kA (@160 Vdc)
I nominal	50 A

Field wiring terminal marking



Attention:

These products are intended for operation within circuits not connected directly to the supply mains (galvanically isolated from the supply, i.e. on transformer secondary).

Attention:

Integral solid state short circuit protection does not provide branch circuit protection.

Branch circuit protection must be provided in accordance with the National Electrical Code and any additional local codes.

Attention:

CSA: For Canada: Integral solid state short circuit protection does not provide branch circuit protection. Branch circuit protection must be provided in accordance with the Canadian Electrical Code, part I.

Attention:

Operation of this equipment requires detailed installation and operation instructions provided in the Installation/Operation manual intended for use with this product. This information is provided online at www.synapticon.com/documentation.com and can be downloaded as PDF.

A hard copy of this information may be ordered at +49 7031 / 30 478 -0

L'utilisation de cet appareil nécessite des instructions détaillées d'installation et de fonctionnement fournies dans le manuel d'installation/utilisation prévu pour ce produit. Cette information est disponible en ligne sur www.synapticon.com/documentation.com et peut être téléchargée sous forme PDF.

La version papier peut être commandée au +49 7031 / 30 478 -0

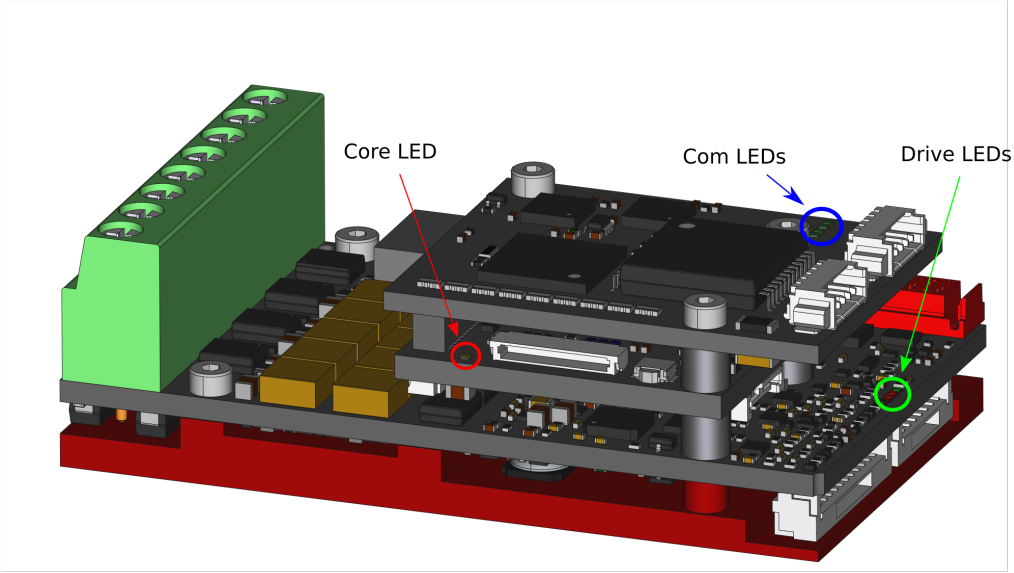
Note:

Motor overtemperature sensing is not provided and must be provided in the field.

LED signals

Overview

SOMANET Node has several LEDs that show the servo drive's state or indicate errors.



Legend

Off
Constantly luminous
Single flashes (1 pulse per second)
Slow blinking (3 pulses per second)
Fast blinking (10 pulses per second)
Two colored behavior: One part of the LED is constantly luminous, the other is blinking slowly (3 pulses per second)
Two colored behavior: One part of the LED is constantly luminous, the other is blinking fast (10 pulses per second)

Core LED

The tri-color LED is used by the firmware to indicate internal status and errors. It provides an overview of the firmware state at a glance to quickly see whether a drive is operating nominally.

Firmware

	CiA 402 State						
	Not Ready to Switch On	Switch on Disabled	Ready to Switch On	Switched On	Operation Enabled	Quick Stop	Fault and Fault Reaction

EtherCAT State	BOOT							
	INIT							
	PREOP							
	SAFEOP							
	OP							

Bootloader

Signals that are shown when the Bootloader is active:

Bootloader State	No Fault (firmware exists)	Fault (no firmware exists)
Idle		
Operation in progress		

> Note:
> If your servo drive is running Bootloader v1.0 you may find the LED fading in rainbow colors indicating that the bootloader is active.

Drive LED

The four LEDs on the Drive module show Hardware Errors.

LED 1: Red	LED 2: Red	LED 3: Red	LED 4: Green	Explanation
				No error
				Watchdog ticks fault
				Dead-time fault Phase A
				Dead-time fault Phase B
				Dead-time fault Phase C

LED 1: Red	LED 2: Red	LED 3: Red	LED 4: Green	Explanation
■		■		Dead-time fault Phase D
	■	■		Over-current in Phase A / Phase B / DC-bus
■	■	■		Over-voltage / Under-voltage / Over-temperature

Details about these errors and remedies can be found here: Error Report Object

Com LED

Signals on the Com LED are required for EtherCAT conformity.

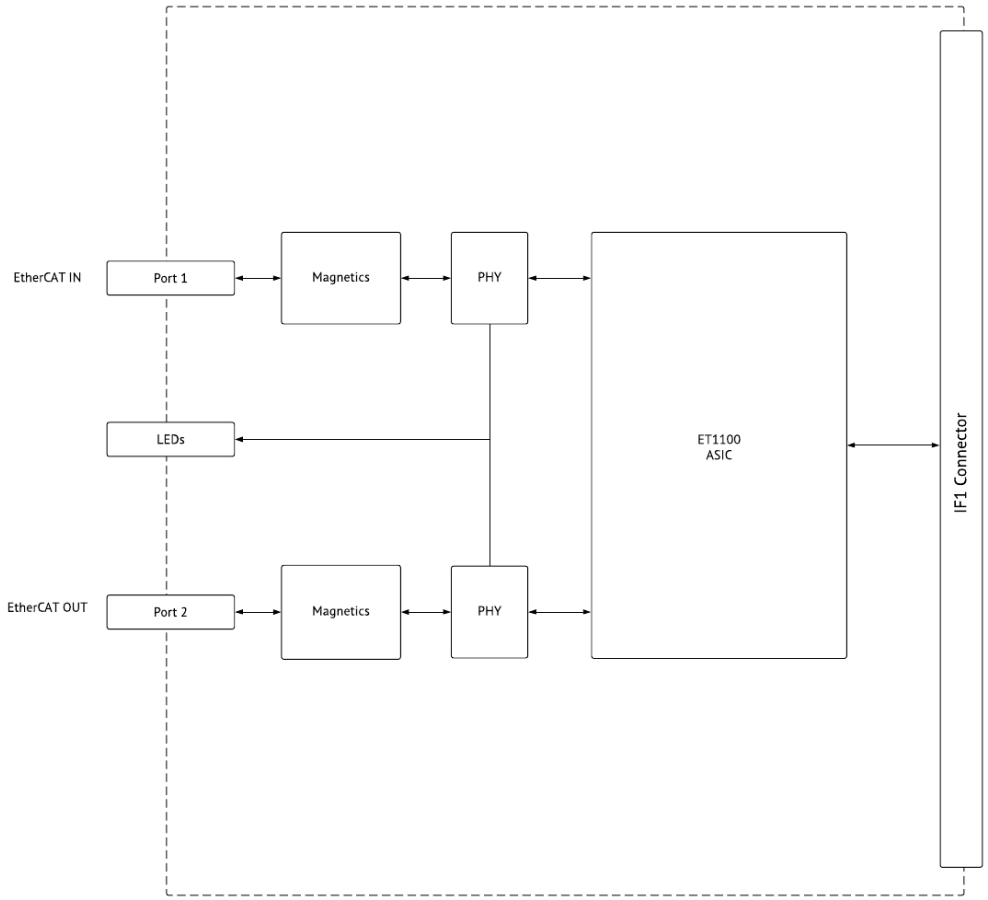
LED 1: OUT: ACT	LED 2: OUT: LINK	LED 3: IN: ACT	LED 4: IN: LINK
▨	■	▨	■

LINK is constantly showing a green light if there is a cable connected to either OUT or IN.
ACT is blinking when there are transmissions on either OUT or IN.
If none of the LEDs are active, check if the EtherCAT cables are connected properly.

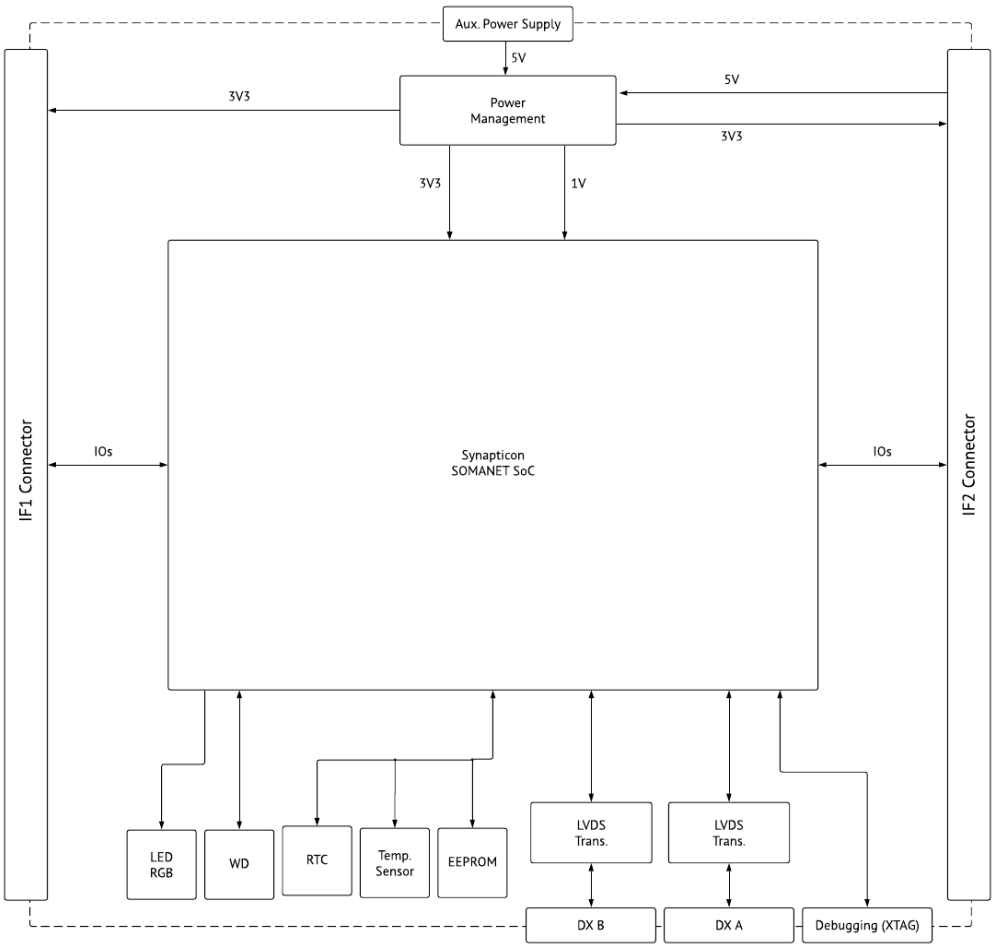
Hardware Diagrams

Due to the modular nature of SOMANET Node, the hardware diagrams of each module are listed separately.

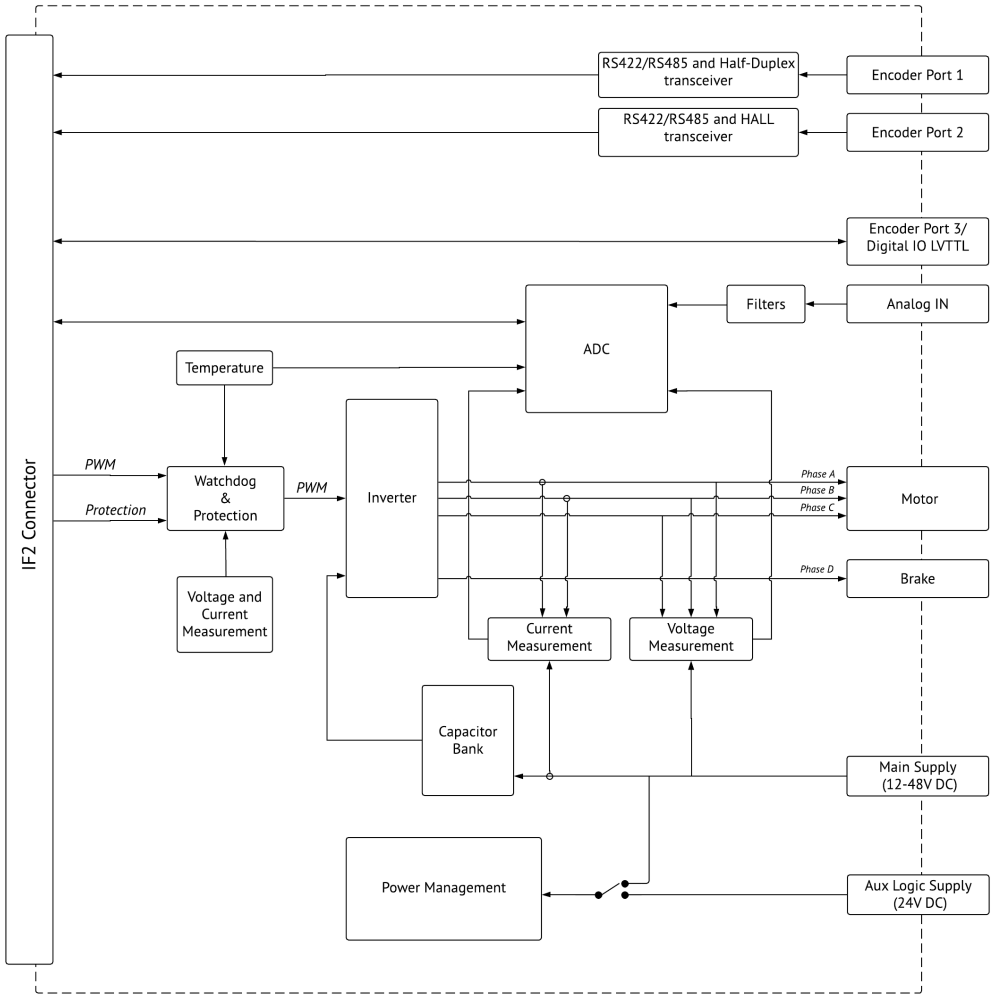
EtherCAT module



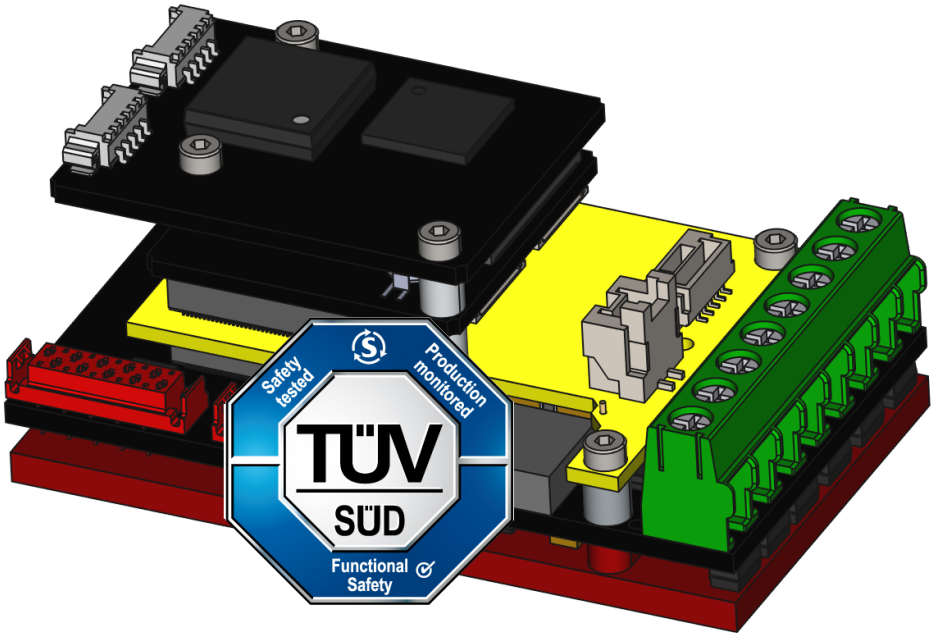
Processor module



Drive module



SOMANET Node Safety (Original instructions)



SOMANET Node Safety provides two drive integrated safety functions according to IEC 61800-5-2:2017 as a pure hardware solution

1. STO (Safe Torque Off) prevents the creation of torque in the motor and enables an uncontrolled stop in accordance with stop category 0 of IEC 60204-1.

2. SBC (Safe Brake Control) provides a switchable (PWM) voltage output (0 V - U_bus) to directly and safely control an external motor brake.

Attention:

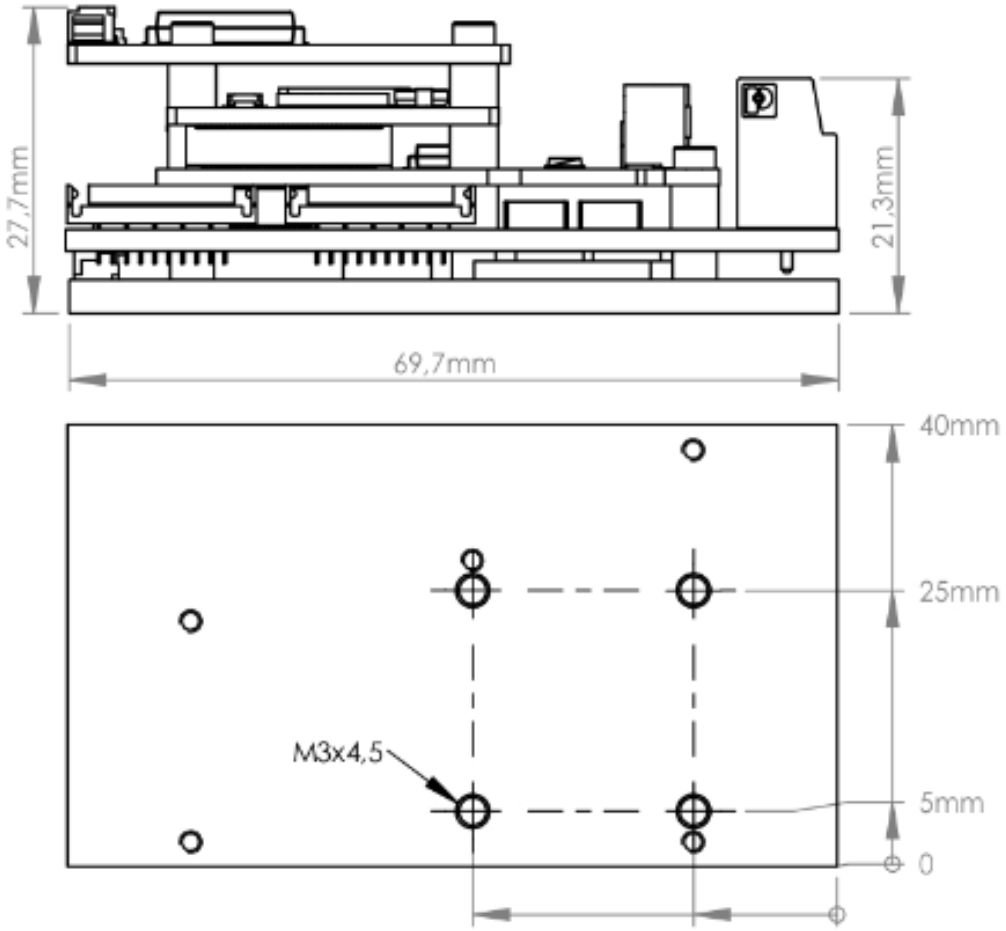
SOMANET Node Safety safely inhibits currents to the motor and to a brake when STO-SBC command has been given (e.g. from an emergency stop device).

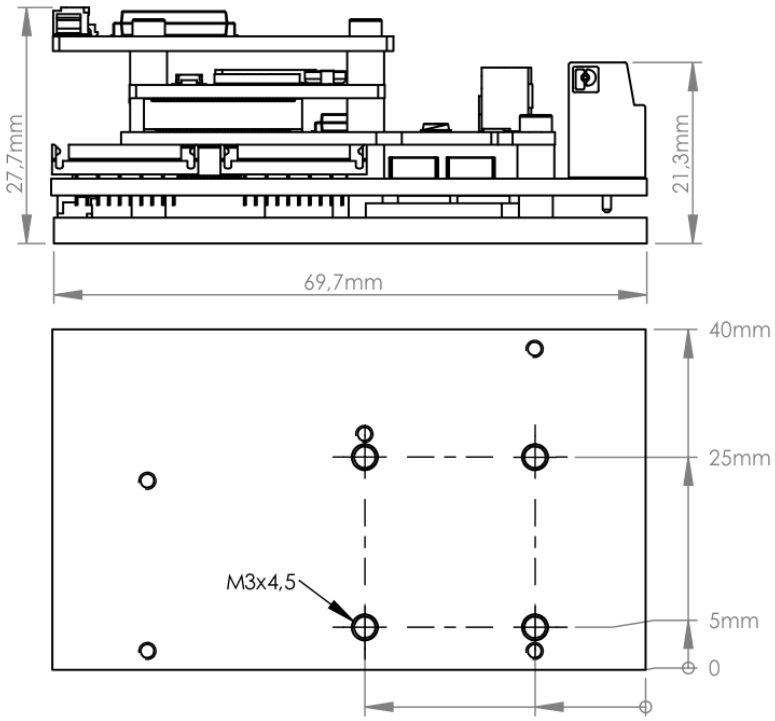
When the command has been released (e.g. manually resetting the emergency stop device) the currents are allowed to be carried to the motor and to the brake.

The safety system must be designed to meet the requirement “restarting the machinery after a stoppage must be possible only by an intended actuation of a control device provided for the purpose”.

- Specifications of the safety functions
- Setup of the safety functions
- Using the safety functions
- Examples for realising safety functions
- Commissioning and maintenance
- Changelog of the safety-related documentation

Dimensions





Specifications of the safety functions

System requirements

Required Firmware

- v4.2.0 or higher

Supported Hardware

- SOMANET Node 2000 Safety (E revision or newer) *
- SOMANET Node 1000 Safety (E revision or newer)
- SOMANET Node 400 Safety (E revision or newer)

* External fuse at main supply (48 VDC) is required for Node 2000 Safety dimensioned according to driven load.

Technical specifications

Safety parameters	
Safety Integration level (SIL) according to IEC 61508:2010	SIL 3
Performance Level (PL) according to ISO	PL e cat. 3

Safety parameters	
13849-1:2015	
Safe Failure Fraction (SFF)	99.99 %
Probability of dangerous failure per hour (PFH_d)	$1.07 \cdot 10^{-10}$ /hour
Probability of dangerous failure on demand (PFD_{avg})	$9.41 \cdot 10^{-6}$
Common Cause Failure (CCF)	>65
Hardware Fault Tolerance (HFT)	1
Reaction time	10 ms
De-activation time	10 ms
Mean Time to dangerous Failure ($MTTF_d$)	3035 years (capped to 100 years)
Diagnostic coverage (DC_{avg})	99%
HW-type	Type A
Mission time T_M	20 a
Supported safety modes	High, Low

Geometrical properties	
Dimensions	70 · 40 · 27.7 mm
Weight	96 g (incl. Heatsink)

Installation requirements	
Max. installation altitude	2000 m

Installation requirements	
Pollution Degree	PD 2
Supported Motors	1 x 3-phase BLDC-Motor and 1 active brake

Attention:
Servo drives including STO-SBC module shall be used only in Pollution Degree environments class PD 2 according to IEC 61800-5-1:2007. During normal operation only non-conductive pollution may occur, temporarily occurring conductivity caused by condensation may only be expected when the servo drive is out of operation.
PD2 environment can be reached for example with environmentally controlled location which does not contain conductive contamination or by using IP54 enclosure according to IEC 60529:1989 + A1:1999 + A2:2013.

Safe digital inputs (STO-SBC input)	
Number of digital inputs	2 (dual-channel safety input for activating STO-SBC function)
Digital input type according to IEC61131-2:2007	Type 1
Voltage range	0-60 VDC
Max. transient input voltage	80 VDC
Input voltage, logic 1	15-60 V
Input voltage, logic 0	0-5 V
Input current	Max. 6 mA (@24 VDC)

Safe digital inputs (STO-SBC input)	
Test pulse toleration (OSSD pulses)	Max. 1 ms
Reverse voltage protected	Yes
Allowed discrepancy between digital inputs	100 ms See safety diagnostic section

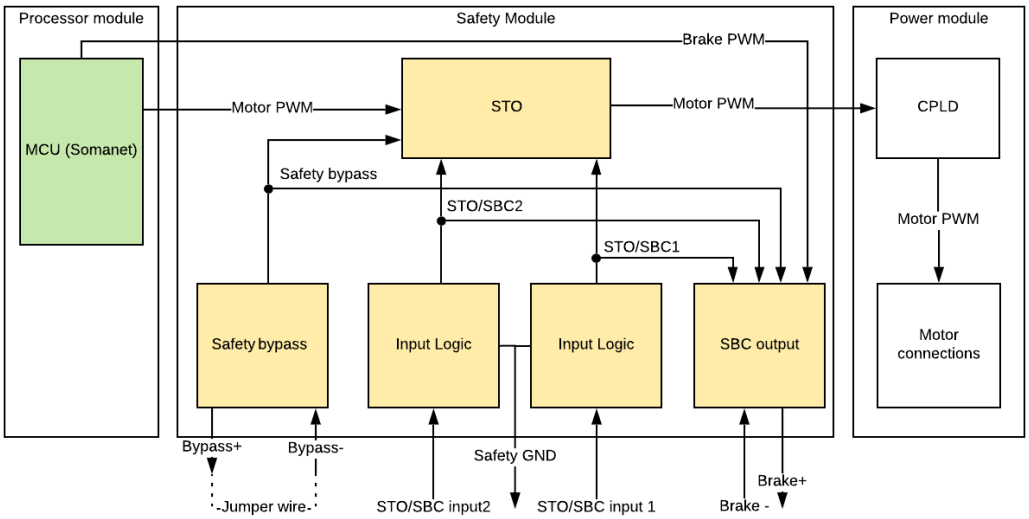
Safe brake output (SBC)	
Number of digital outputs	2 (dual-channel safety output for one brake system)
Max. output voltage	Servo drive main power supply (48 VDC)
Min. output voltage	0 V
Max. current.	1 A
Max. peak current.	4 A

Note:

This PWM controlled brake is configured by THE SAME OBJECT as the regular brake.

Block diagram

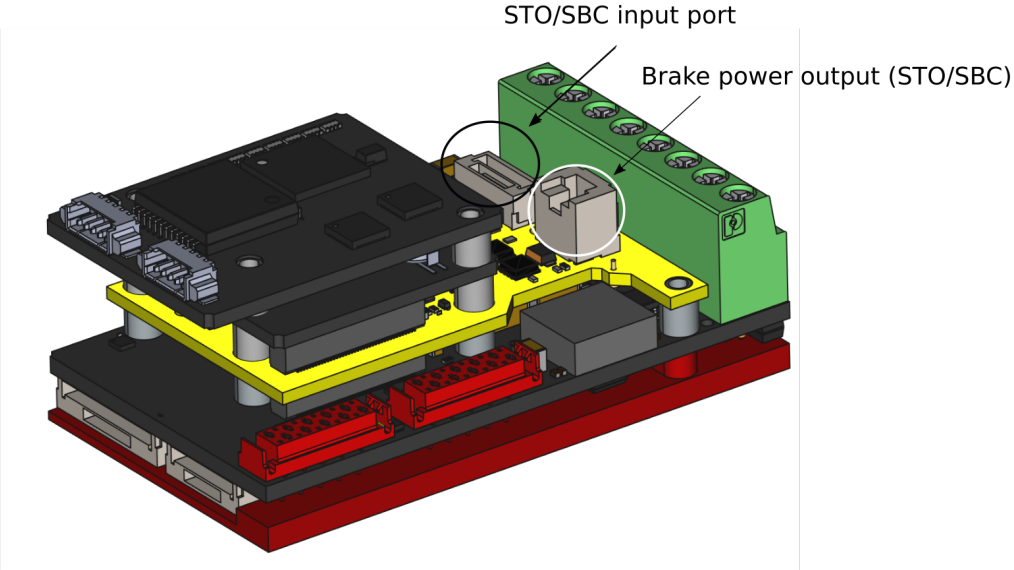
Simplified block diagram of SOMANET Node Safety:



For details of the connectors please refer to:

[Safety connectors]

Setup of the safety functions



Attention:

For SIL 3, PL e, cat.3 it is required to send test pulses from the PLC to the output or to ensure fault exclusion for wiring “Short circuit between any two conductors” according to ISO 13849-2:2012 between signals STO-SBC 1 and STO-SBC 2 permanently connected (fixed) and protected against external damage, e.g. by cable

ducting, armouring, OR

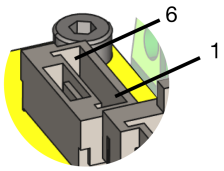
separate multicore cables, OR

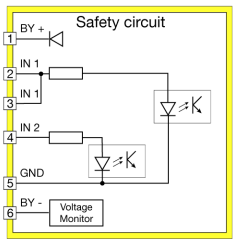
within an electrical enclosure, OR

individually shielded with earth connection

Connecting the STO/SBC inputs

STO/SBC input port





Pin#	Designation	Function	Description
1	BY+	Bypass+	Connect to BY- (Pin 6) to bypass STO/SBC
2	IN1	STO/SBC input 1	Input 1 for STO/SBC
3	IN1	STO/SBC input 1	Extra Pin for Input 1
4	IN2	STO/SBC input 2	Input 2 for STO/SBC
5	GND	Safety GND	Must be isolated from main Ground
6	BY-	Bypass-	Connect to BY+ (Pin 1) to bypass STO/SBC

Wiring examples for the STO/SBC inputs

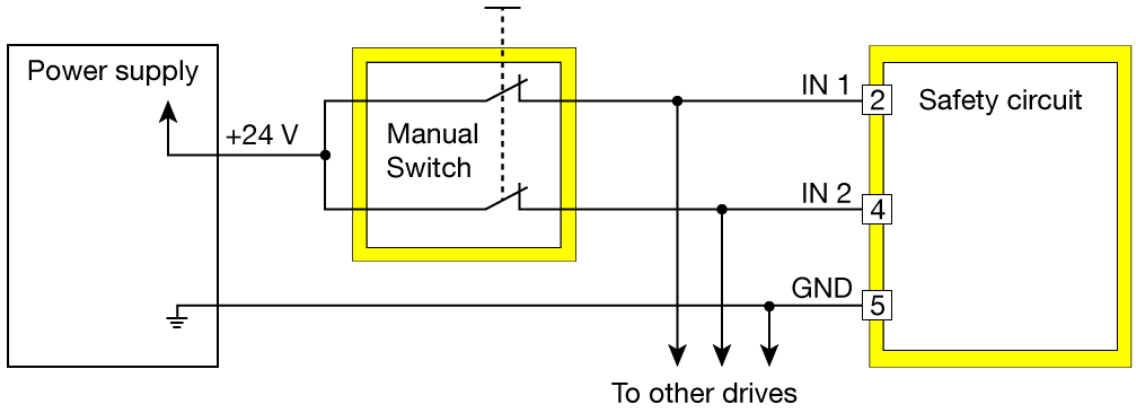
The wiring difffers depending on the safety device used:

- Manual switch covers any emergency device that can be connected directly to the safety input of the servo drive.
- If safety PLCs are used it is important to distinguish between plus-plus and plus-minus logic
    - PLCs using plus-plus logic must be connected to both inputs for redundancy
    - PLCs using plus-minus logic are only connected to the input 1 which must be bridged to input 2. The redundant channel in this case is the pulsed safety ground signal.

> Important:
>
> Please note that Safety GND is an isolated signal from drive GND.

Manual switch

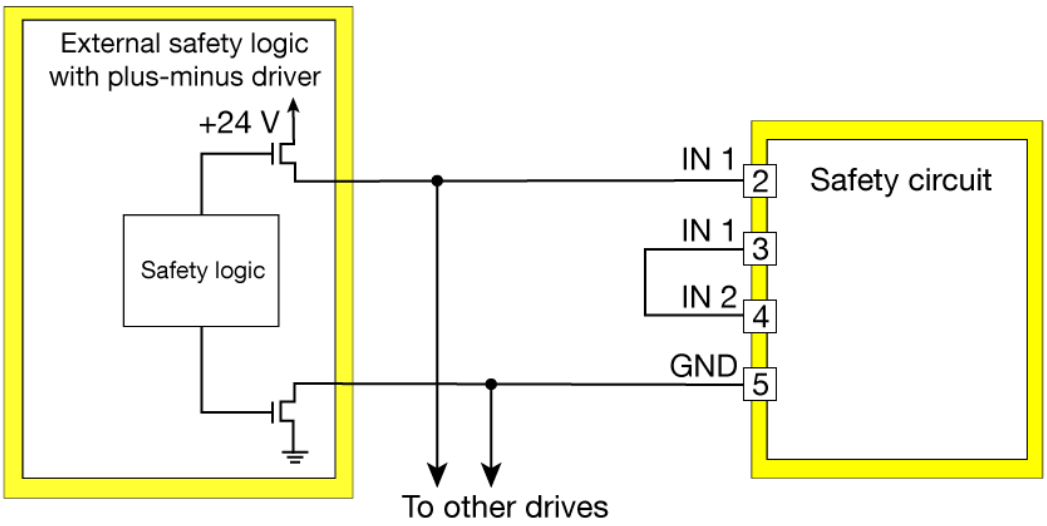
Allows safety functions up to **SIL 3, PL e, cat.3.**



Safety PLC (plus-minus output)

Allows safety functions up to **SIL 3, PL e, cat.3.**

> Note:
>
> Safety GND functions as a secondary line when a PLC with plus-minus output is used.



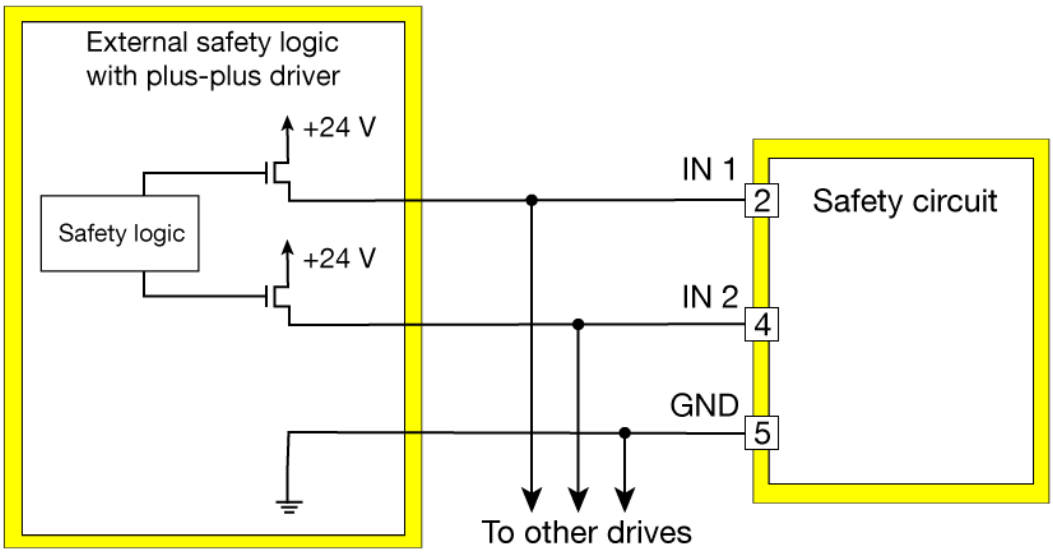
Attention:

In case of multiple drive connection, ensure that the output driver has sufficient current driving capability.

Safety PLC (plus-plus output)

Allows safety functions up to **SIL 3, PL e, cat.3.**

For example: Safety PLC output or safety sensors (e.g. Safety light curtain, safety scanner etc.)



Attention:

In case of multiple drive connection, ensure that the output driver has sufficient current driving capability.

Connection diagram for the brake

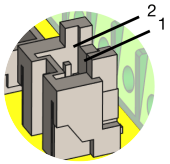
Attention:

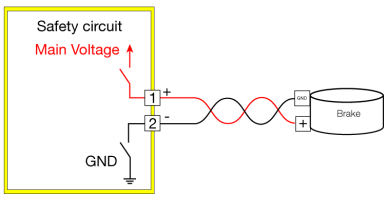
Phase D can’t be used for connecting a brake when Node Safety is used:

- use the safety brake connector of the safety circuit (SBC out) instead of phase D of the drive circuit.

- ensure to connect the brake GND signal to the Brake- signal of the safety circuit (SBC out) instead of the general GND.

Brake power output (STO/SBC)





Pin #	Designation	Function
1	+	Brake +
2	-	Brake -

Note:

The safe brake is PWM based and can be configured with the regular brake object.

Connector mating parts and cable lengths

Connector description	Manufacturer	Mating part number	Crimping Contact	Cable
Safe brake port	JST	ZER-02V-S	SZE-002T-P0.3	0.08-0.21 mm² / 28-24 AWG
Safe digital I/O port	JST	GHR-06V-S	SSHL-002T-P0.2	0.05-0.13mm² / AWG 30-26

Please observe the maximum allowed cable lengths:

Connection	Maximum wire length
STO-SBC	30 m

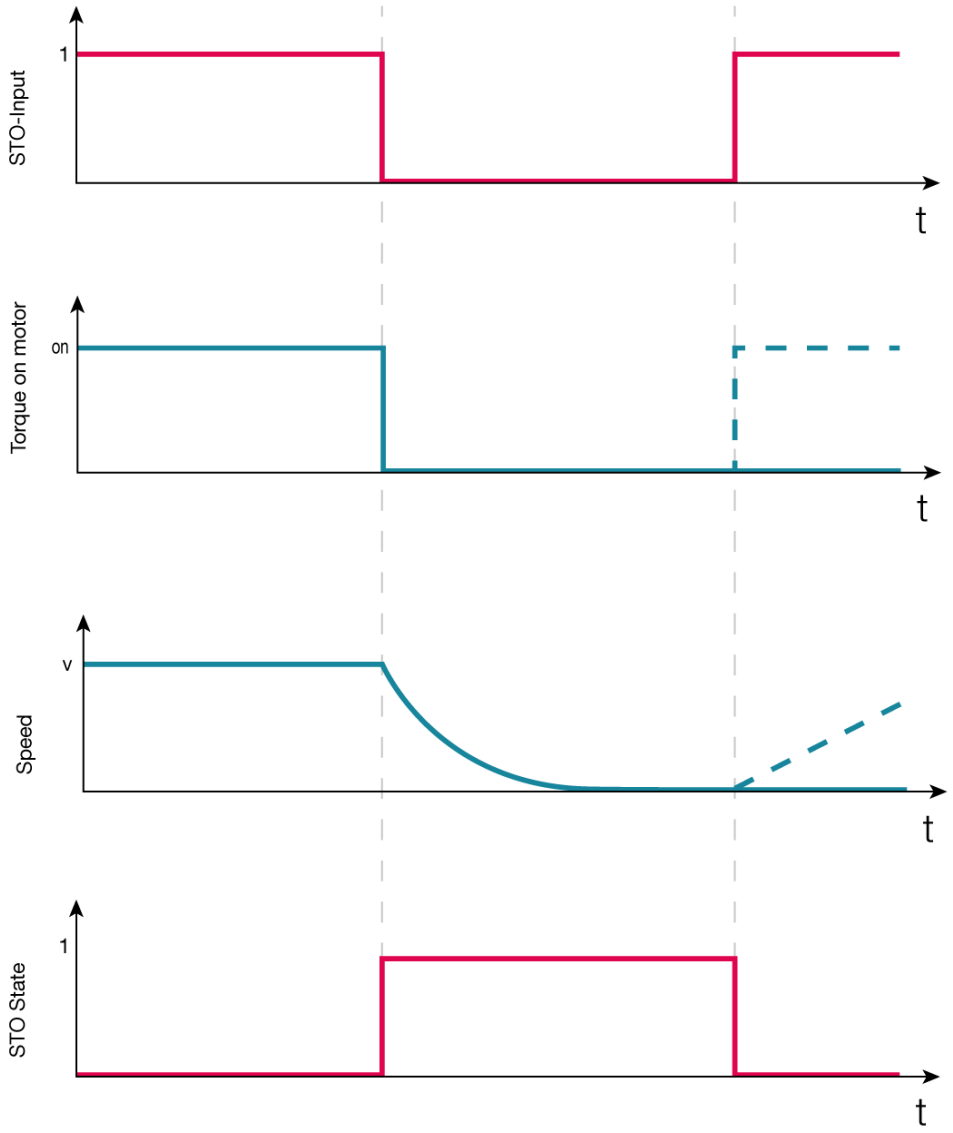
Using the safety functions

Attention

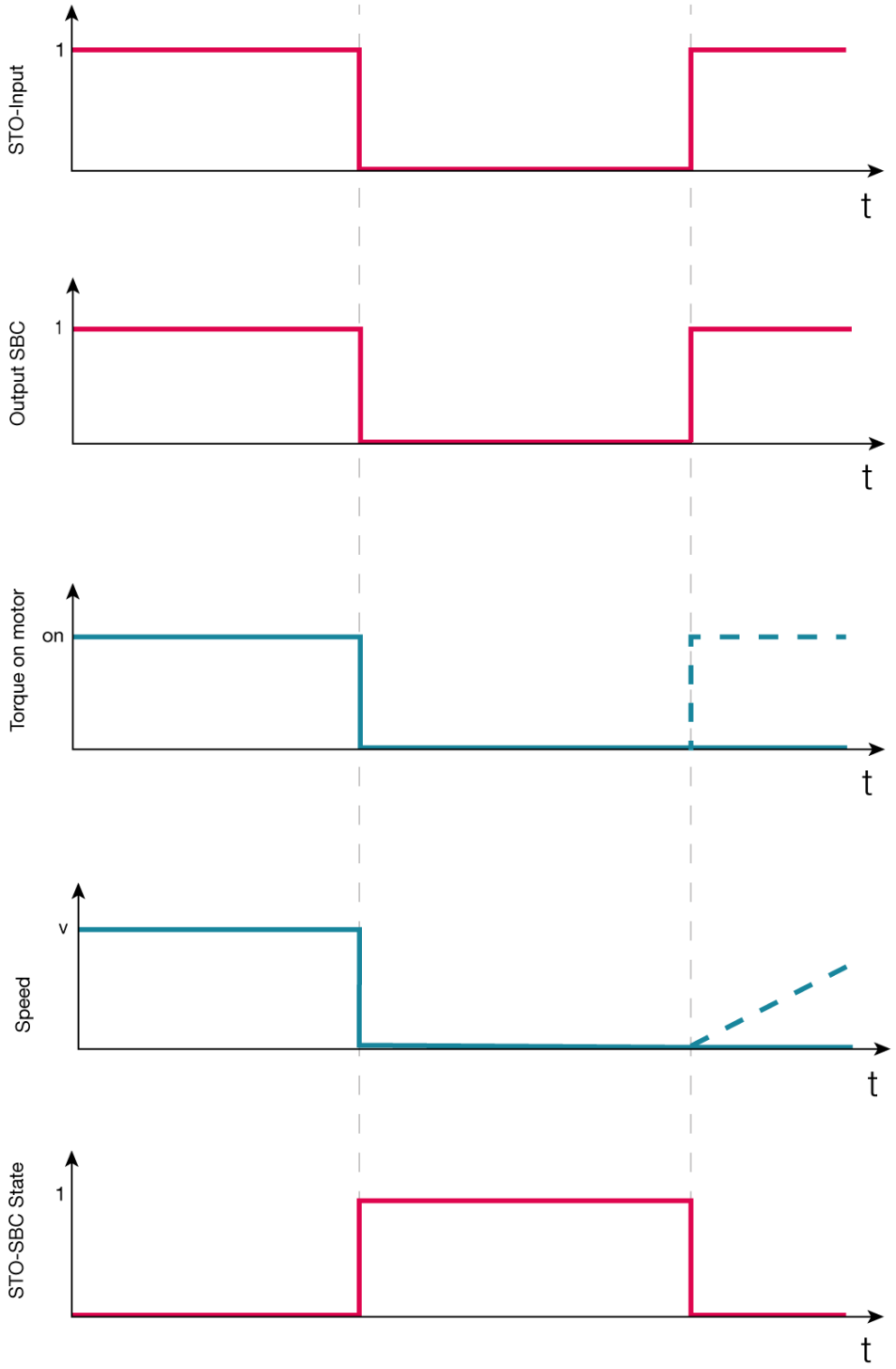
Activating SBC independently of or with a delay to STO is not possible.When STO is active SBC will also be activated, if a brake is connected it will be engaged.
To prevent damaging the brake, a Safe Stop Category 1 (SS1) is recommended.

Timing diagrams

Timing diagram for STO-function without connected brake



Timing diagram for STO-function with SBC-function



Truth table for digital inputs

Digital input STO-SBC 1	Digital input STO-SBC 2	Internal fault	Safety Status-word	Error entry
0	0	no	1	
0	1	no	1	“SfeDilvd”
1	0	no	1	“SfeDilvd”
1	1	no	0	
0	0	yes	1	“SfeFault”

Diagnostic functions

Fault diagnostics

The safety circuit has two integrated diagnostic functions:

- Comparing safety digital input statuses. The fault is activated after 100ms discrepancy of the STO-SBC inputs. In case a fault is detected, the servo drive will enter *Fault state* and indicate a fault “SfeDilvd” in the Error Report Object.
- During the activation of the STO-SBC function the servo drive verifies that the two channels of the module are internally operating correctly. In case an internal fault is detected, the servo drive will enter *Fault state* and indicate a fault “SfeFault” in the Error Report Object.

Fault reaction

In case a fault has been detected, the drive will stay in fault state until manual reset has been given and the brake is closed.

Resetting diagnostic faults

To reset a fault condition, please:

- request STO/SBC (for example by activating the emergency stop device) and
- reset the fault

Attention:

When the fault resetting was not successful, power cycle the servo drive and investigate the reason for the failure.

STO-SBC status register

The *Safety Statusword* indicates the current state of safety functions and *Safety Digital Input Diagnostics* shows the state of the Safety Module inputs.

Index	Name	Descriptions
0x6621	Safety Statusword	Subindex: 1 - Bit 1: STO status is the first bit of byte 2 - Bit 2: SBC status is the first bit of byte STO/SBC status: 0: is not active 1: is active
0x2611	Safety Digital Input	Subindex: 1 - Input 1: 2 - Input 2: Input 1/2: 0: Input is low 1: Input is high

Important:

Using the SBC function while running the motor may damage the brake due to mechanical stress. During normal operation it is recommended to activate STO-SBC after the motor

has come to a halt e.g. by introducing a delay with an external safety logic device such as a safety timer. This way an equivalent of SS1 with time monitoring is implemented.

Attention:
The STO-function does not provide electrical isolation from the mains supply. If electrical changes need to be carried out on the system (e.g. modifying the motor cabling), the servo drive shall be completely isolated from mains supply with a mechanical switch.

Examples for realising safety functions

The following examples can be used as a guidance for implementing some of the most common safety functions. A risk and hazard analysis is required for each application to find necessary safety functionalities and the requirements for the safety functions.

Prevention of unexpected start-up

STO and SBC function can be used when power removal is required to prevent an unexpected start-up according to ISO 14118. A lockable safety switch for activating the STO-SBC function is required.

The function “Prevention of unexpected start-up” can be used for system maintenance activities (e.g. Repairing/cleaning activities inside hazardous areas)

Stop category 0 (STO)

The STO and SBC functions can be used for implementing the category 0 emergency stop function according to IEC 60204-1:2016.

The Emergency stop function can be realized

- With an emergency stop switch connected directly to control the STO-SBC input signals
- With an emergency stop relay + emergency stop switch.
- With safety PLC + emergency stop switch.

Please read the section about the working principle of the Safety circuit.

All cases above require correct behaviour from the controlling PLC. After releasing the emergency stop device, a separate and deliberate action must be used to restart the system (restarting can be done with a non-safety PLC).

Attention

IEC 60204-1:2016 requires “reset shall not initiate a restart” regarding emergency stop.

Stop category 1 (SS1)

This example depicts the prerequisites and wiring for an emergency stop that fulfills the requirements according to

- IEC 60204-1:2016 stop category 1 (a controlled stop with power available to the machine actuators to achieve the stop and then removal of power when the stop is achieved)
- ISO 13849-1:2015 PLe, Cat. 3
- IEC 61508:2010 SIL3
- IEC 61800-5-2:2017 SS1 (Safe stop 1)

When the Emergency Stop Device (such as a push button or a light curtain) signals the execution of an emergency stop, two redundant sets of contactors are triggered:

- K1/K2 gives immediately the signal to the Master PLC/IPC Digital Input. When the Digital Input is logical 0, the Master PLC/IPC shall request “quick stop” via EtherCAT Controlword 0x6040.
- K3/K4 are triggered after a configurable time delay. During the time delay, the machine can be gracefully controlled to a stop before the activation of the SOMANET Node STO-SBC function.

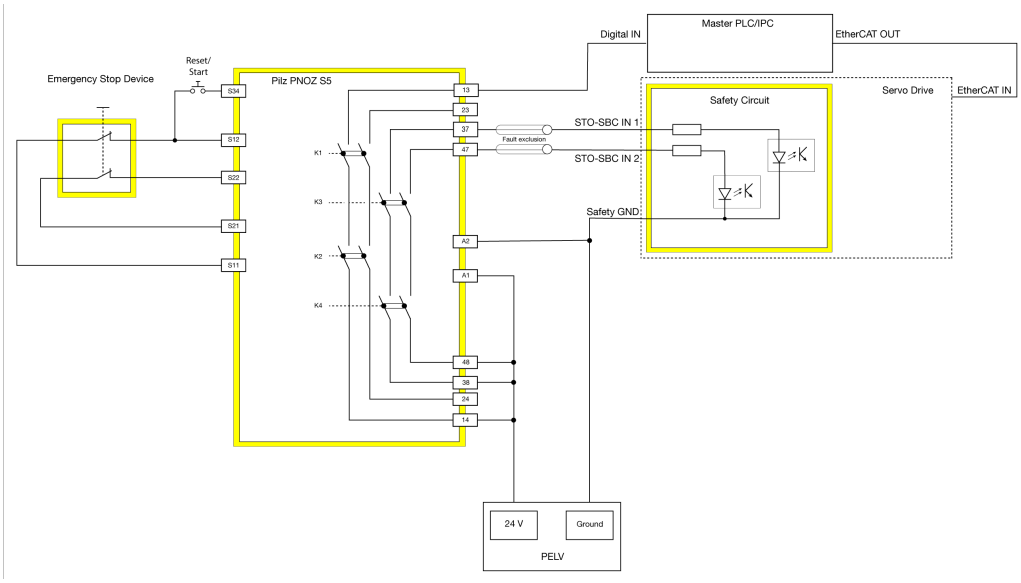
Prerequisites

- SOMANET Node Safety
- Safety Relay
- Emergency stop device
- Power supply (PELV)
- Reset/Start switch

Note

It is necessary to use a **momentary switch** for Reset/Start function to fulfill the requirement regarding emergency stop “The reset of the emergency stop command **shall not restart** the machinery.”

Wiring



Attention:

For SIL 3, PL e, cat.3 it is required to send test pulses from the PLC to the output or to ensure fault exclusion for wiring “Short circuit between any two conductors” according to ISO 13849-2:2012 between signals STO-SBC 1 and STO-SBC 2

permanently connected (fixed) and protected against external damage, e.g. by cable ducting, armouring, OR

separate multicore cables, OR

within an electrical enclosure, OR

individually shielded with earth connection

Configuration

- **Master PLC/IPC**Program the digital input to request quick stop via EtherCAT Controlword 0x6040
- **Delay time (in Safety relay)**Configure the time according to your application. The delay time should be longer than the maximum deceleration time of the application.
- Ramp time (in SOMANET)Configure the Quick stop deceleration (0x6085) according to your application.

Verification

Calculate the PFHd (high demand mode) or PFD (low demand mode) of the whole safety function with the values given by the device’s safety manual. Verify that the PFD/PFH_d of the safety

function is sufficiently low for the desired safety level. You can find the values for SOMANET Node Safety here.

Commissioning and maintenance

Commissioning

Before the system is put into operation the safety functions must be tested/validated.

The STO-SBC function shall be activated/deactivated and the reactions shall be monitored (the reaction depends if used as emergency stop or prevention of unexpected start-up or another safety function).

1. Safety wiring diagrams and the commissioning test shall be documented and stored (who,what,when)
2. Define maintenance routines according to the need (see chapter maintenance)

Maintenance

IEC 61800-5-2:2017 requires to have a maintenance test for STO-SBC function. The frequency of testing depends on the required safety level:

- one test per year for SIL 2 , PL d / category 3
- one test per three months for SIL 3 , PL e / category 3

The maintenance test can be done by activating STO/SBC function through STO-SBC inputs and observe the reaction of the system.

Changelog of the safety-related documentation

Version	Reason	Details	Date released
2.4.3	Initial release Safety Module A.3	Original Safety Module Documentation	2019-11-26
2.4.5.2	Update	Add safety values according to EN ISO 13849 and IEC 61508 Add changelog	2019-12-10
2.4.5.4	Update	Update safety specs and wiring diagrams	2019-12-17

Version	Reason	Details	Date released
		Add supported motors	
2.4.5.5	Update	Update safety specs	2019-12-19
2.4.5.6	Update	Add Common Cause Failure	2020-01-09
3.0.0	Update	Add Example section Add example “Wiring for a Safe Stop 1”	2020-04-06
3.0.1	Update	Remove Feedback Output from table	2020-04-15
3.1.1	Update	Correct truth table	2020-05-15

Commissioning and Tuning Guide

SOMANET OBLAC Drives guides the user through the setup process, supports configuration and tuning of the SOMANET servo drive and makes the operation and first steps in turning a motor as easy as possible.

OBLAC Drives setup There are two ways to run SOMANET OBLAC Drives:

OBLAC Box is a physical machine that comes preinstalled with Linux OS and runs OBLAC Drives

OBLAC Drives virtual machine comes preinstalled with OBLAC Drives. It can be run on Windows, Linux and macOS

Troubleshooting If something goes wrong while using OBLAC Drives you can find troubleshooting instructions here

Using OBLAC Drives Overview of the functionality

Update or downgrade OBLAC Drives Install and run any version of OBLAC Drives that fits your drive

Set up your servo drive Explanation of the most important parameters

Test your drive at the playground Rotate your motor to a target position, at a target velocity or apply a target torque

Motion control tuning Instructions on how to perform manual and automatic tuning of the controller gains

Automatic tuning Overview of the Autotuning feature

System Identification This feature allows to generate a precise model of the mechanical system with one click

Position control loop Automatically generates gain values for the cascaded position controller

Velocity control loop Automatically generates gain values for the velocity controller

Manual tuning Description of steps required for manually tuning the controller gains

Manual Tuning of the current controller In the torque control loop the current is controlled, while the torque is directly proportional to the current

Manual tuning of the velocity control loop Instructions for manually tuning the proportional and integral gains of the velocity control loop if the drive should be operated in velocity control mode

Manual tuning of the position control loop Instructions for manually tuning the cascaded PID controller

OBLAC Drives Setup

There are two ways to run SOMANET OBLAC Drives:

OBLAC Box: is a physical machine that comes preinstalled with Linux OS and runs OBLAC Box.

OBLAC Drives Virtual Machine: comes preinstalled with OBLAC Box. It can be run on Windows, Linux and macOS.

If you run into any problems, this might be ale to help you:

Troubleshooting: If something goes wrong while using OBLAC Drives you can find troubleshooting instructions here

OBLAC Box

> Important:
>
> In case you are not using OBLAC Box, please refer to our OBLAC Drives virtual machine installation guide.

OBLAC Box



OBLAC Box 2

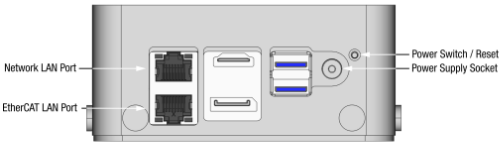


OBLAC Box is a physical machine that comes preinstalled with Linux OS and runs OBLAC Drives and the related services. It supports access over WiFi or local area network.

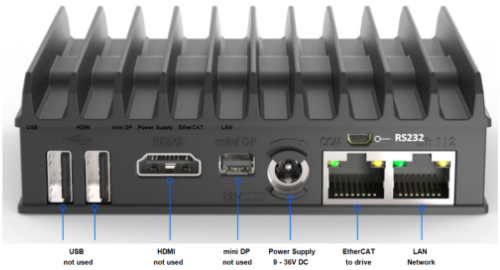
WiFi Frequency	2.4/5 GHz (Antenna, Dual Band 2.4/5 GHz, Rubber Duck, RP-SMA)

Connectors

OBLAC Box



OBLAC Box 2



Installation

OBLAC Box

> Note:
>
> The power supply has an output rating of 5 V_{DC}, 4000 mA.

OBLAC Box 2

> Note:
>
> The power supply has an output rating of 12 V_{DC}, 3000 mA. OBLAC Box 2 can be supplied with 9-36 V_{DC}, ≥15 W

1. Plug in the power connector; the box will automatically boot.
2. Connect one or multiple drives to the EtherCAT port 1 on the box
3. Establish a network connection to the box by Wi-Fi or per Network Cable (see Network Connection to the OBLAC Box)
4. Open a browser (Google Chrome preferred) and connect to the box (see Open OBLAC Drives)
5. The OBLAC drives dashboard will appear
6. Update the box with the latest available firmware (see Update OBLAC Drives)
7. All connected SOMANET drives should be visible. Select the Servo Drive you want to configure and tune

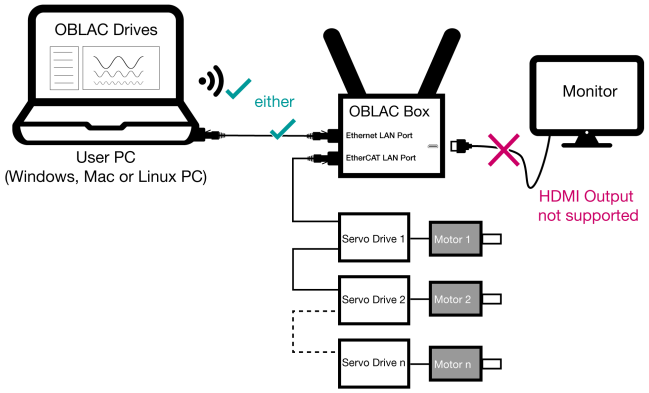
Attention:

Ensure that the LAN ports are used correctly: The servo drives have to be connected to the EtherCAT LAN port. If the ports are mixed up, no communication to the drives and no connection to the box via LAN is possible.

Note:

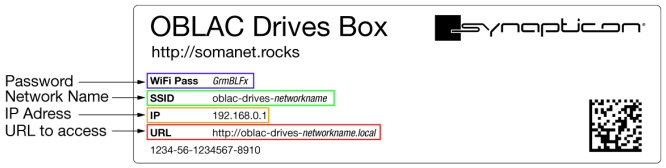
The HDMI port is without function. You require a dedicated host PC to control OBLAC Box.

Multiple instances of a browser connection to a box can be established. A box can operate multiple connected drives independently, but one browser tab can only operate one drive. You can even connect to one drive on multiple browser instances (e.g. operate and get diagnostic information at the same time). But the last action will override the previous. There is no interlock between the instances.

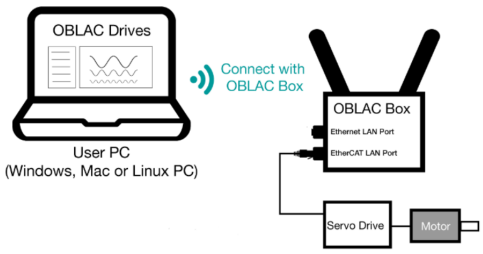


Network Connection to the OBLAC box

Depending on different needs, several ways to establish a network connection are possible. In each case the drives have to be connected to the EtherCAT port of the box. Some important device specific information is written on the sticker placed on the box:

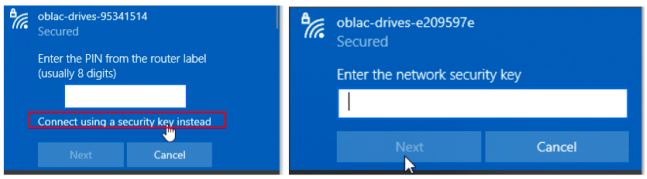


Connection Host-PC ↔ Box by Wi-Fi



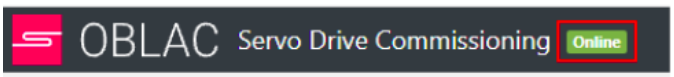
> Note:
>
> This is the preferred way to connect to an OBLAC Box.

To establish the connection you have to activate the Wi-Fi on your host PC and connect to the Wi-Fi with the network name written on the box. Each box has a unique SSID. Open the Wi-Fi configuration of your PC, select the given network and enter the password. On Windows 10, you have to enter the password as security key:



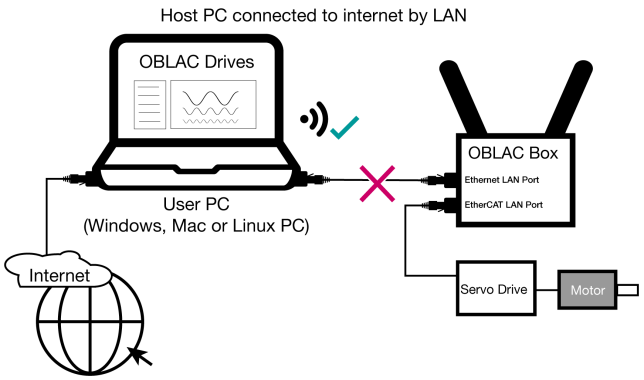
After the Wi-Fi connection is established, you have to open a browser (Google Chrome preferred) and enter the IP address given by the label in your browser's address bar (always 192.168.0.1). Alternatively you can enter the URL given on the label of the box. The OBLAC Drives dashboard will appear. You can now operate connected drives with OBLAC Drives.

To be able to do any updates, a connection to the internet or a previously saved bundle is required. Different ways to realize this connection are described in the following chapters. A connection between the box and the internet will be reported by the green "online" sign in the top line of the OBLAC Drives page.



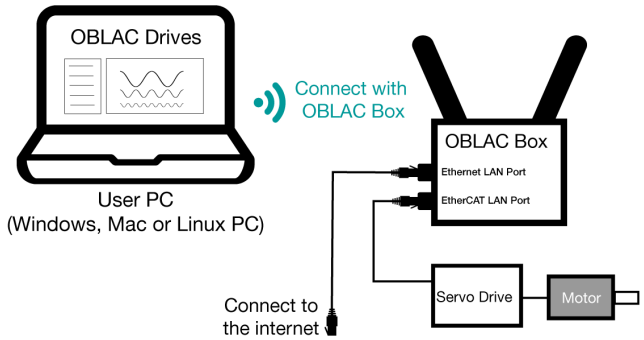
Host-PC connected to Internet by LAN connection

If the host PC is connected to the internet by a LAN connection or a second Wi-Fi connection, the box will automatically connect to the Internet through the host PC.



Box connected to internet by LAN connection

If there is no connection to the internet with the host PC possible, you can alternatively connect the OBLAC box to a network by LAN network port on the box. On this port, DHCP is configured and a DHCP server is required to give the box an IP-address.



> Attention:
>
> In some companies, unknown devices will not get an IP address from the server out of security reasons. In this case this will not work. Ask your IT/network responsible if you fail to get online with the box.

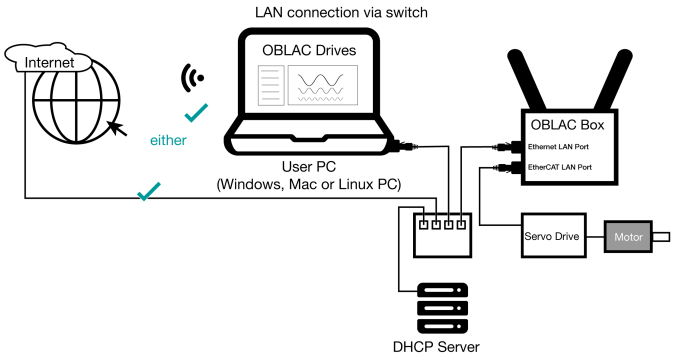
Host-PC and Box not connected to Internet

If neither the PC nor the box can't be connected to the internet, the offline update by bundles can be used (see Offline Updates).

Connection Host-PC ↔PC by LAN cable

If the Wi-Fi connection of the host PC is already used for accessing the company network / internet connection, the connection to OBLAC Drives can be done by connecting the host PC to the OBLAC box by a LAN cable. Use the LAN / Network port in this case.

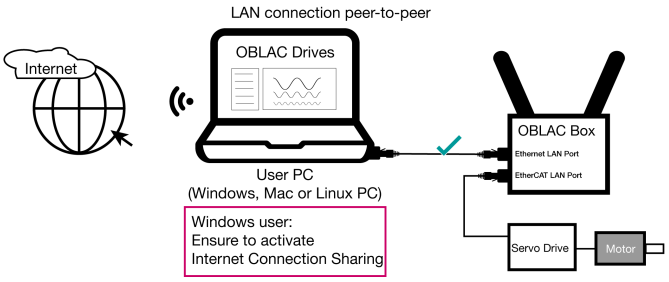
LAN connection via Switch and connection to a network with DHCP server



If the box and the PC are connected to a LAN that includes a DHCP server (regularly by connecting to a switch), the box and the PC port will get an IP address from the DHCP server in the same range. To connect to OBLAC Drives, it is not needed to know this IP-Address. It is enough to call the local URL of the Box printed on the label of the box (http://oblac-drives-xxxxxxxx.local)

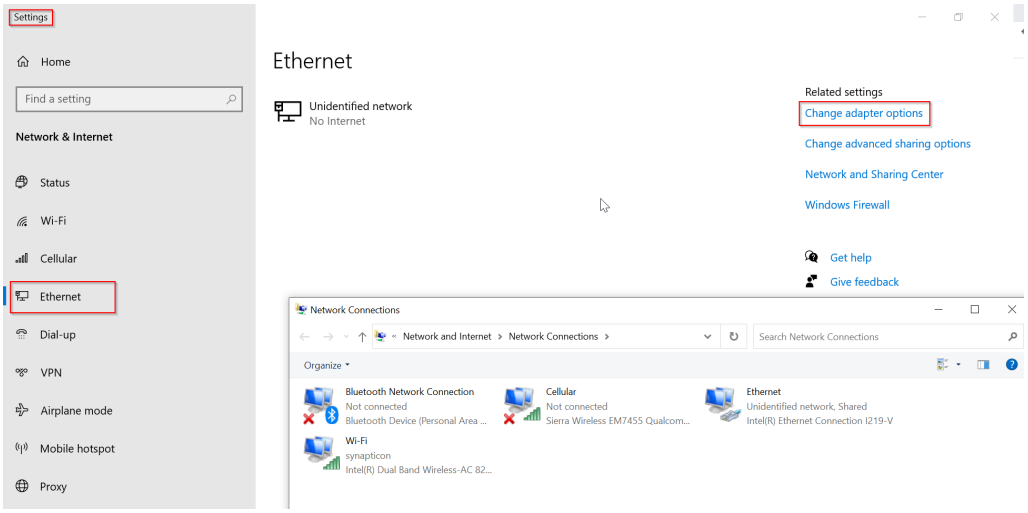
> Attention:
>
> In some companies, unknown devices will not get an IP address from the server out of security reasons. In this case this will not work.

LAN connection Peer to Peer

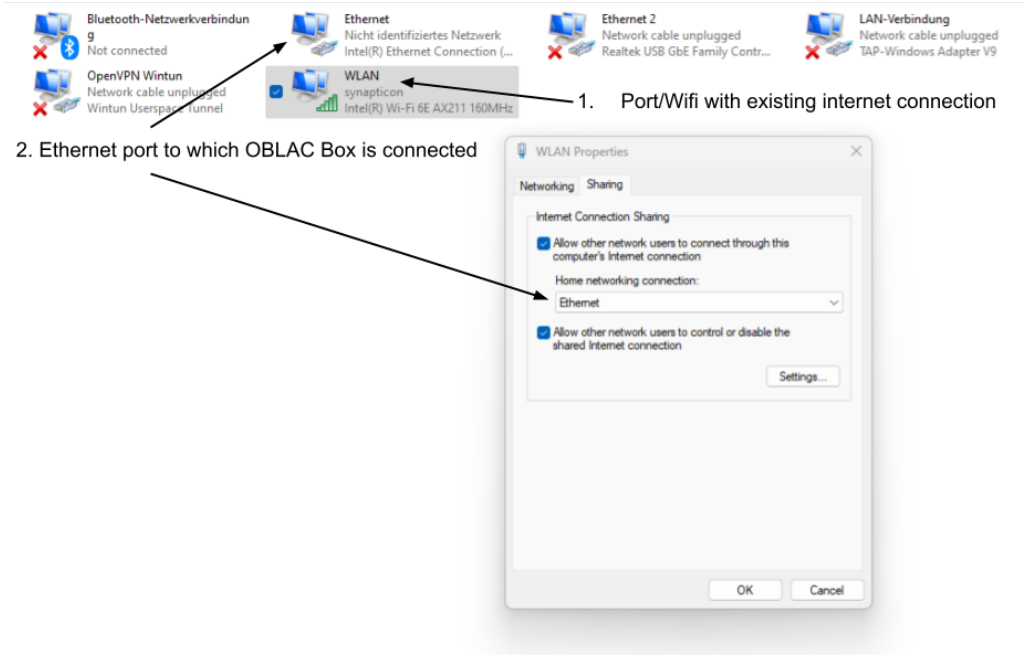


If there is no DHCP server available, a single peer to peer connection can be used. Just plug in an Ethernet patch cable between your PC board and the LAN port of the OBLAC box. But in addition ICS (Internet Connection Sharing) has to be enabled on the PC.

Go to Windows Settings - Network & Internet:



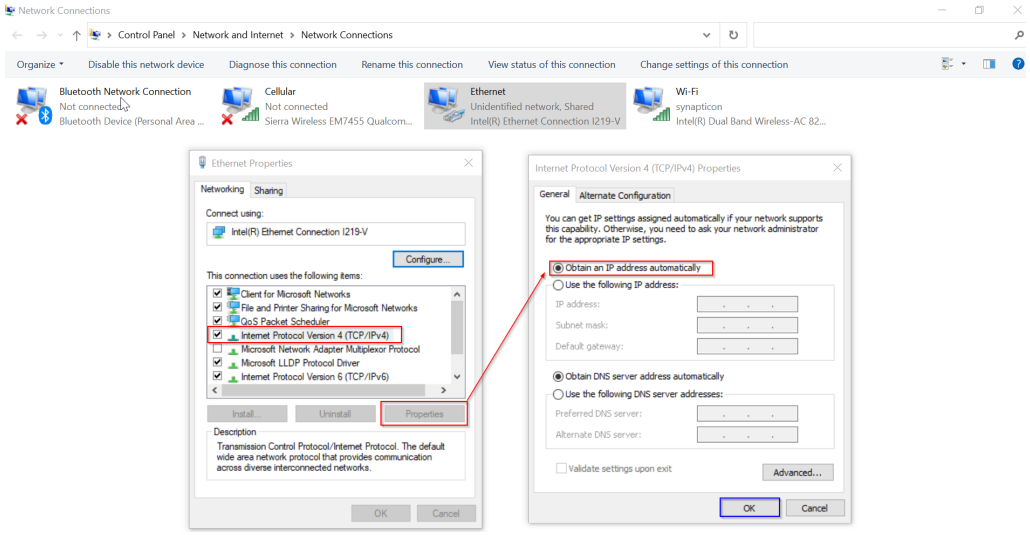
- Right-click on the Ethernet Port you are connected to the internet.
- Open properties and Sharing tab.
- Enable the first check box.
- Select the Ethernet port your OBLAC box is connected to.



Now you can connect to the box and open OBLAC Drives out of the browser by connecting to http://oblac-drives-xxxxxxxx.local. You find the box specific address on the label on your box.

The first time after activating the ICS, it might take up to one minute till the bridging is done and you can connect to OBLAC Drives.

The ICS is not working if the IP-address is manually set to a fixed address in the PC. Please ensure that the port is configured to “obtain IP address automatically”.

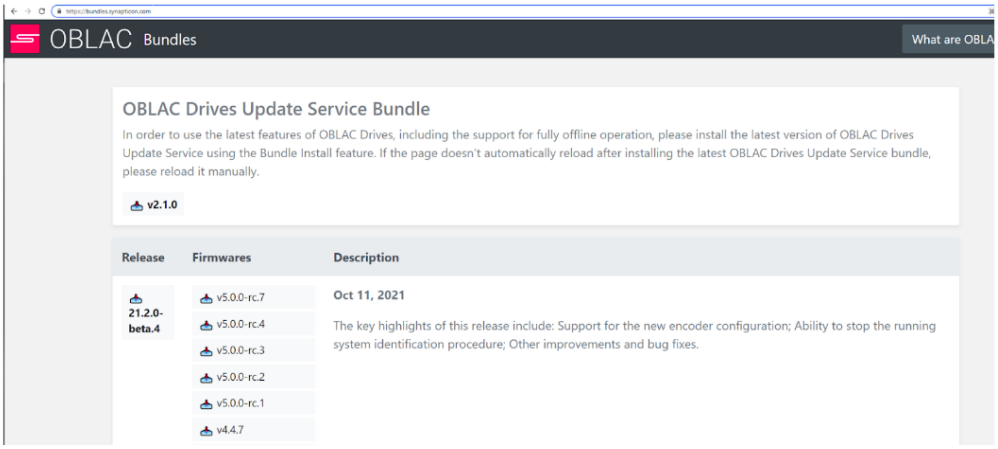


No connection to OBLAC update server

The box retrieves updates from an update server in the cloud. Some company firewalls block those connections. Ask your IT responsible if you get fault messages in not being able to connect to the update server. You can either remove those blockings or continue with offline updates (see below).

Offline Updates

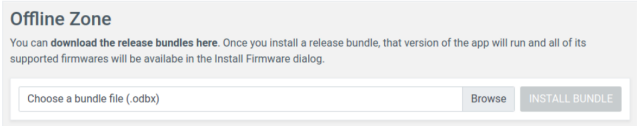
If it is for any reason not possible to get online with the OBLAC box, the offline update with “bundles” is possible. The bundles can be downloaded from https://bundles.synapticon.com/.



There are two types of OBLAC Bundles:

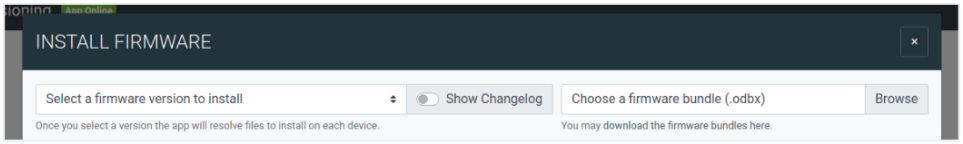
1. Release bundle

   This bundle includes an OBLAC Drives release, e.g. v21.0.0 and all of its supported firmwares. Install it using the Bundle Install feature in OBLAC Drives Update Service. Once installed, the selected release will run, and all of its supported firmwares will be available in the OBLAC Drives Install Firmware dialog.



2. Firmware bundle

   This bundle includes files for all hardware variants of a single firmware version, e.g. v4.4.3. Install it using the OBLAC Drives Install Firmware dialog onto one or more devices.



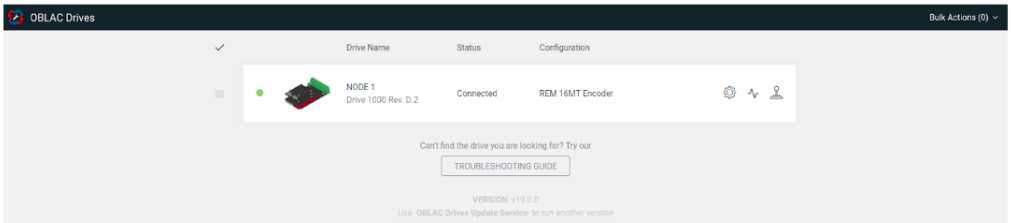
> Note:
>
> Offline update is available from OBLAC Drives version 21.0.1. upwards

Open OBLAC Drives

Open OBLAC Drives in your browser by navigating to http://oblac-drives-networkname.local

> Note:
>
> Windows users need to install the Bonjour Software to open OBLAC Box at the given URL.

If this is not possible you can still access OBLAC Drives via Wi-Fi by entering the IP address given on the sticker. Please check in your browser that the Drive is connected:



Update OBLAC Box

Before using OBLAC Box, you need to install the latest version of OBLAC Drives with our OBLAC Drives Update Service.

It is possible to use the OBLAC Box offline and to install and run the previous versions of OBLAC Drives. Newer versions of OBLAC Drives that get installed will be cached on the box, but they will have to be downloaded from the Internet.

OBLAC Drives Virtual Machine

> Important:
>
> This section is only required if you are **not** using OBLAC Box.

Prerequisite

- OBLAC Drives Virtual Machine is run on a virtualization software. VMware Player version 14 or above for Windows and Linux or VMware Fusion 10 for macOS are needed for running OBLAC Drives Virtual Machine.

> Note:
>
> OBLAC Drives is an application that runs in a Web Browser. We recommend the latest version of Google Chrome.

Getting started

1. Download the OBLAC Drives Virtual Machine image

DOWNLOAD OBLAC DRIVES

If you have trouble downloading from Amazon webservice, please try our Baidu server:

DOWNLOAD OBLAC DRIVES FROM BAIDU

Windows

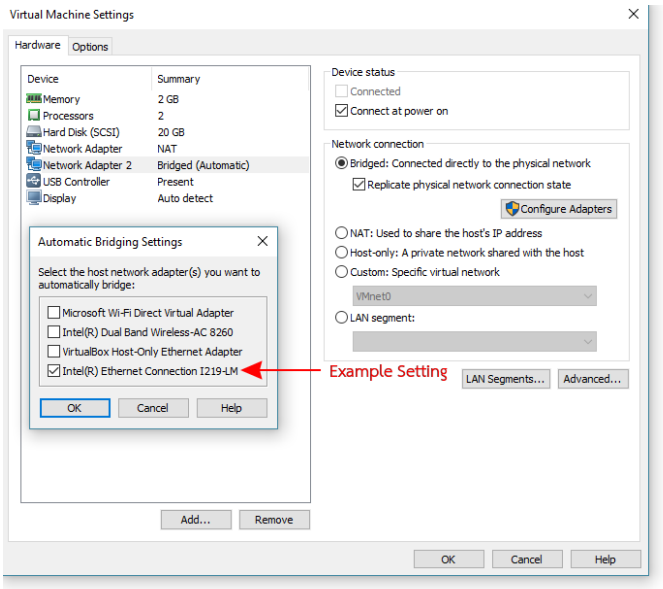
2. Install and open VMware Workstation Player

> Note:
>
> ANY version of VMWare Workstation Player starting from 14.0 can be used for OBLAC Drives.

3. Open a virtual machine
4. Select the downloaded OBLAC Drives image (OVA file) and import it to VMware Player

> Note:
>
> If you get a warning that the import failed, just click “Retry” and it will run.
>
> You may also check “Do not show this message again”

5. Go to Virtual Machine Settings and check if two network adapters are available - one set to NAT and one set to Bridged (Automatic).
   - Ensure that the "Replicate physical network connection state" option is checked.
   - Press OK to save changes.



6. Power up and connect the slaves.
7. Power on the virtual machine

macOS

2. Install and open VMware Fusion

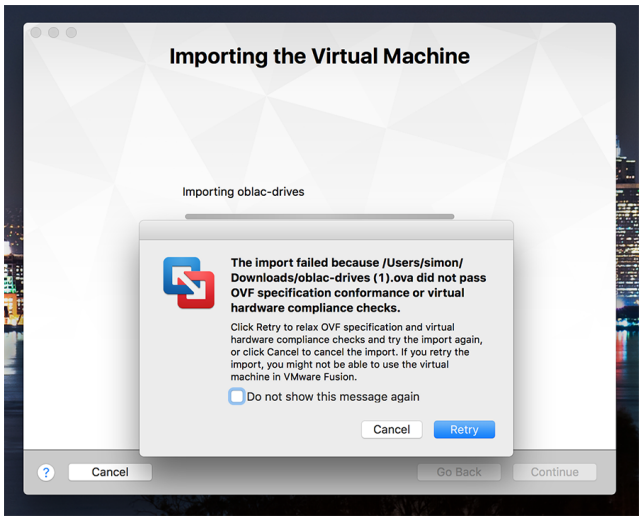
> Note:
>
> ANY version of VMWare Fusion starting from 10.0 can be used for OBLAC Drives.

3. Open the OBLAC Drives image (OVA file) in VMware Fusion

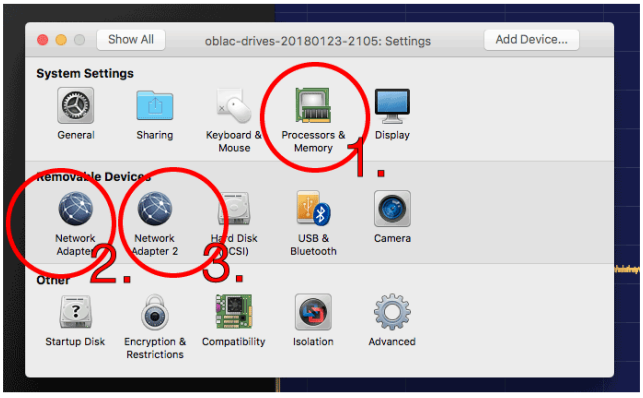
Note:

If you get a warning that the import failed, just click “Retry” and it will run.

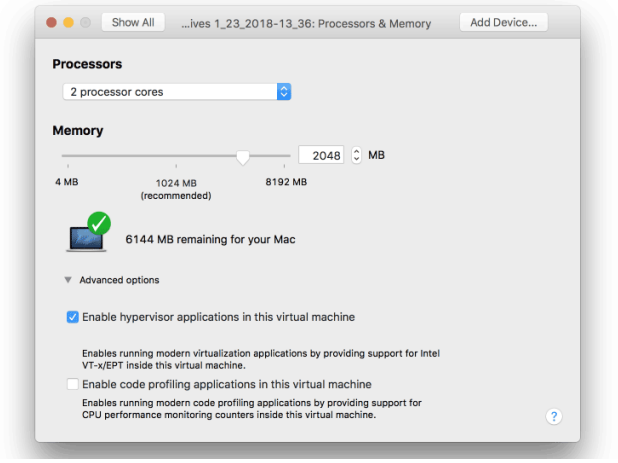
You may also check “Do not show this message again”



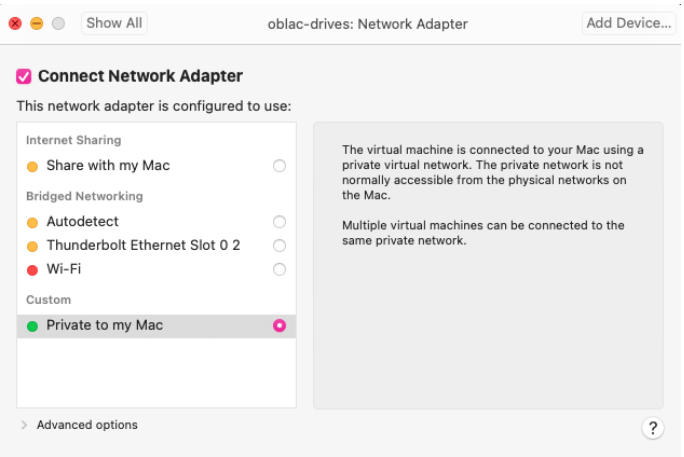
4. Edit the Virtual Machine Settings and set the following (if not already set)



- Set the number of **processor cores** to 2
- Check the “Enable hypervisor application” option



- The **network adapters** should be set to: *Private to my Mac*



5. Power up and connect the slaves.
6. Play the virtual machine.

Linux

2. Install and open VMware Workstation Player

> Note:
>
> ANY version of VMWare Workstation Player starting from 14.0 can be used for OBLAC Drives.

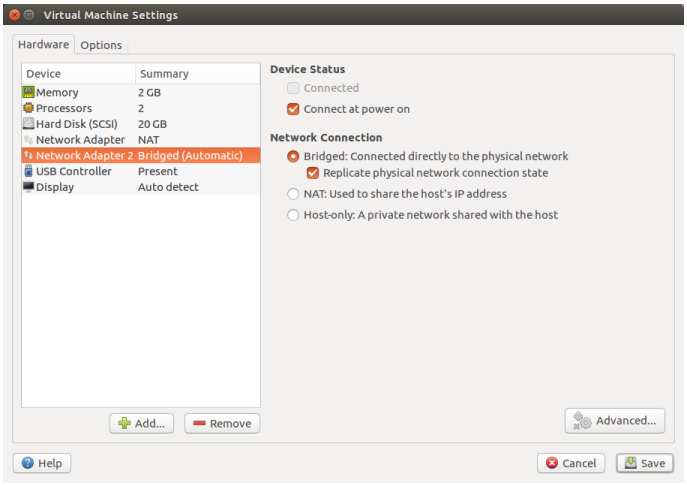
3. Open a virtual machine:

   Select the downloaded OBLAC Drives image (OVA file) and import it to VMware Player

Note:

If you get a warning that the import failed, just click “Retry” and it will run. You may also check “Do not show this message again”

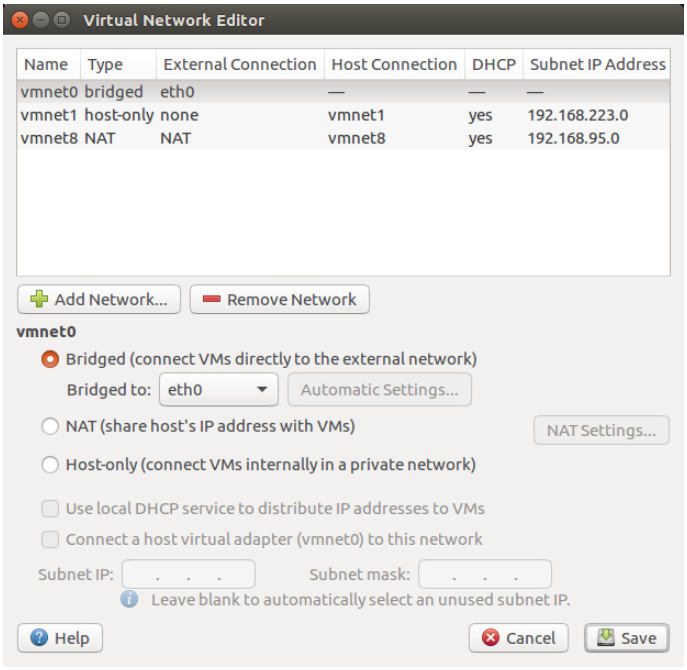
4. Go to Virtual Machine Settings and check if two network adapters are available - one set to NAT and one set to Bridged (Automatic)
   - Ensure that the “Replicate physical network connection state” option is checked
   - Press OK to save changes



5. Network settings
   - Execute

```
sudo /usr/lib/vmware/bin/vmware-netcfg
```

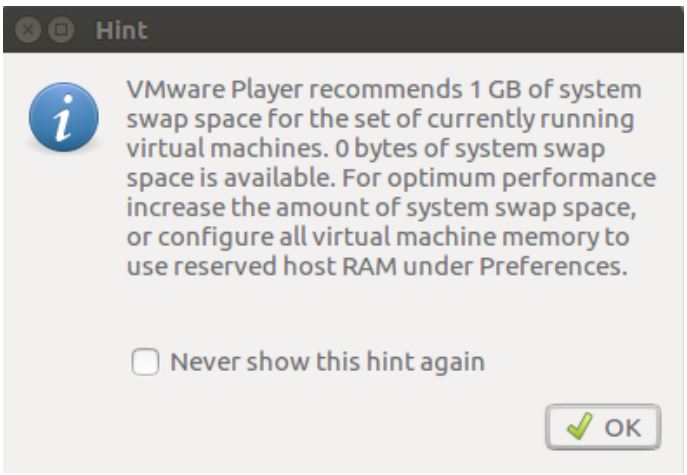
   - Select the bridged network adapter and bridge it to the system network adapter to which the slaves are connected to
     - The network adapter has to be an Ethernet connection, so it likely starts with “eth”
   - Save changes



6. Power up and connect the slaves.

To connect your host PC with the servo drive, simply plug the EtherCAT cable on your Ethernet port (RJ45) of your PC and connect it to the servo drive on EtherCAT port IN.

7. Play the virtual machine
8. In case your systems hints that a swap partition is needed:



instead of creating such a partition you can also use this workaround:

- Close the VMware Player
- Run

```
sudo gedit /etc/vmware/config
```

- Add this line at the end of the file and save:

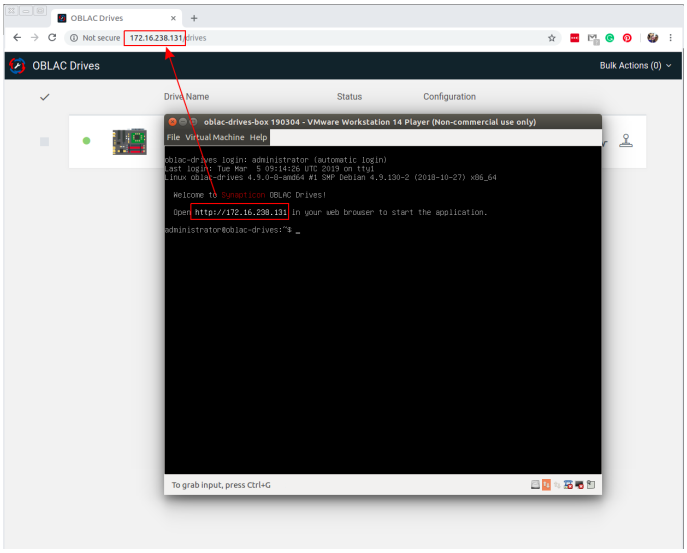
```
prefvmx.minVmMemPct = "100"
```

- at the end of the file and save

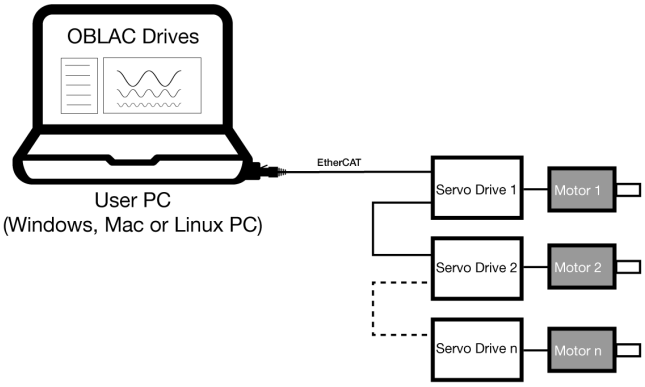
  Restart the VMware Player

Connecting to OBLAC Drives

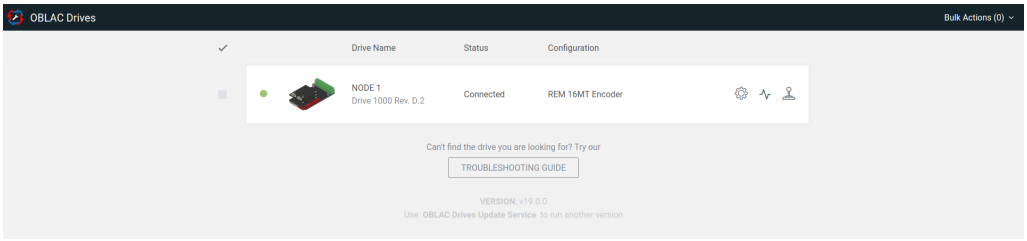
Once your virtual machine is powered up, you will find the IP address of OBLAC Drives at the top of the screen. Enter this IP address in your browser and wait for OBLAC Drives to show the connected drives.



Connect your PC via EtherCAT cable to the servo drive. For tuning, you may connect several servo drives via EtherCAT and tune one by one.



Check in your browser that the Drive is connected:



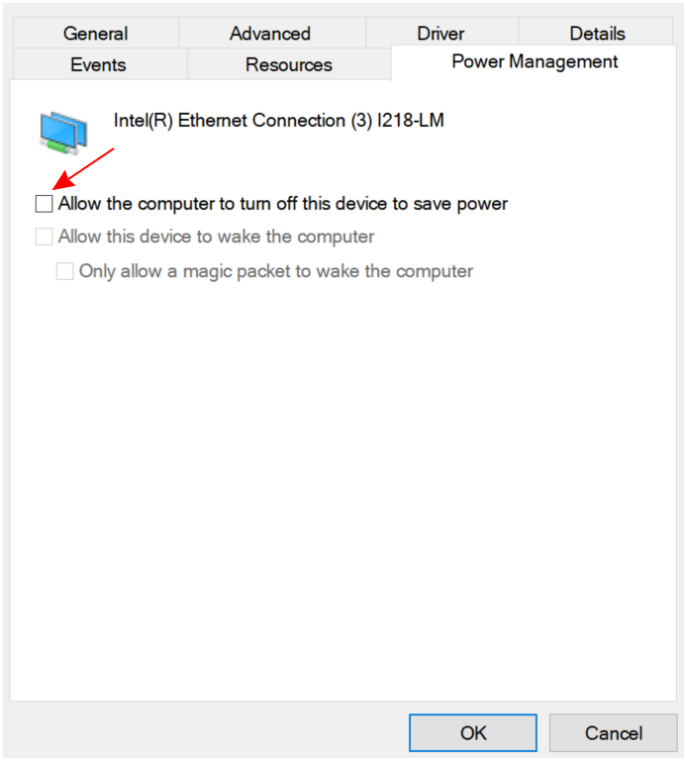
Update OBLAC Drives

Before using OBLAC Drives, you need to install the latest version of OBLAC Drives with our OBLAC Drives Update Service.

It is possible to use OBLAC Drives offline and to install and run the previous versions of OBLAC Drives. Newer versions of OBLAC Drives that get installed will be cached on the box, but they will have to be downloaded from the Internet.

Troubleshooting

1. Always give OBLAC Drives up to a minute to detect the available drives after they have been powered on or connected
2. If the drives aren't found, or anything else seems to be wrong, please try refreshing the web page first and then power cycling the drive(s)
3. If you are running Windows, please ensure that the power-saving functionality for LAN is turned OFF



4. In some cases after the start-up of OBLAC Drives the URL isn't shown and the website can't get accessed. In that case, please try switching the order of the network adapters in the VMware setting for that virtual machine:
    - Bridged (Automatic)
    - NAT

There are more issues listed in our Troubleshooting Guide

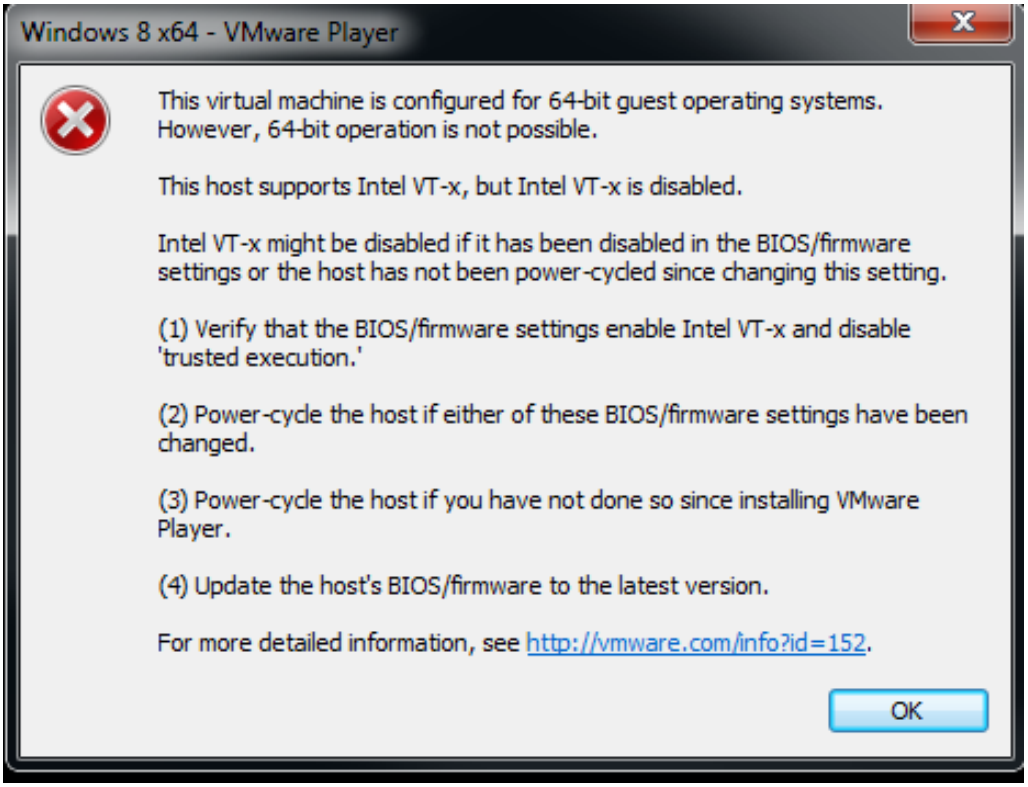
Troubleshooting

This section shall guide you through the most common error scenarios that could occur when using OBLAC Box.

> Important:
>
> Make always sure to carefully read through the setup instructions. A lot of issues can be avoided by strictly following the instructions.

Virtualization not activated

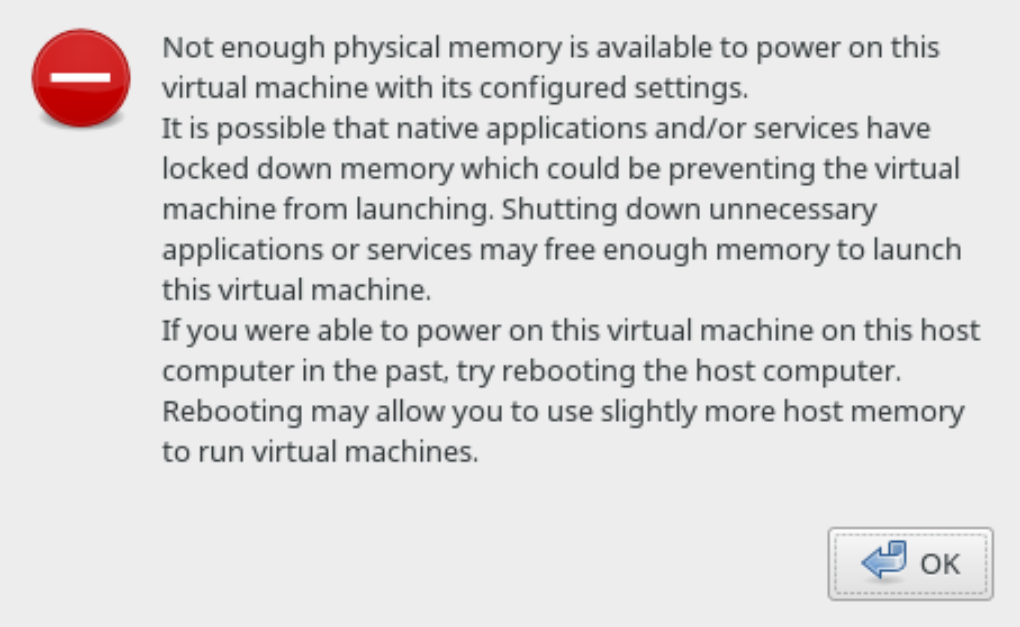
If your computer does not have the virtualization technology activated, OBLAC Drives will most likely not work correctly.



To fix that issue, you have to activate the virtualization technology of your processor (VT-x / AMD-V) in your BIOS.

Memory Warning

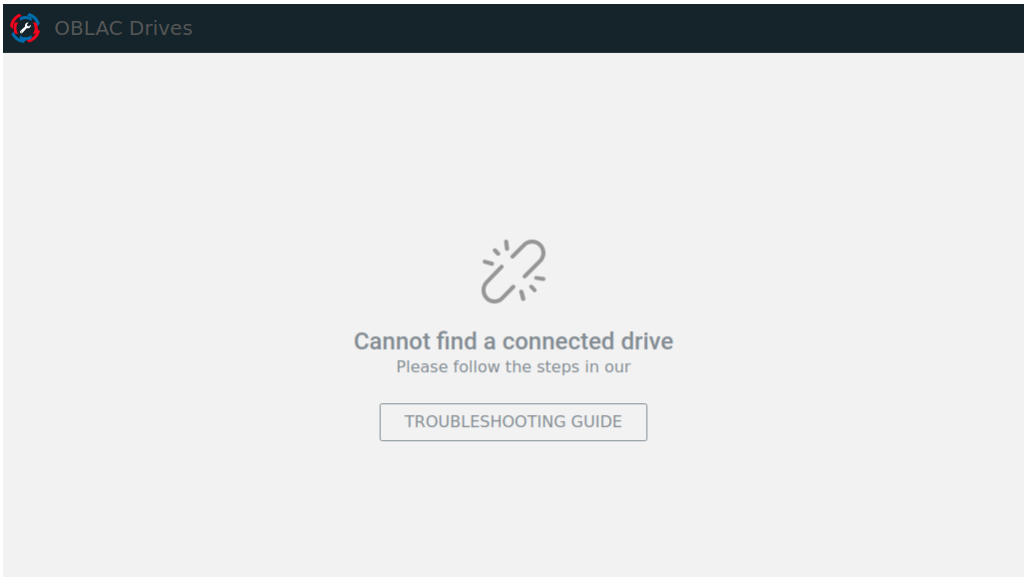
In rare cases your Linux distribution may show the warning: Not enough physical memory available to power on this virtual machine”



To fix this issue, follow the commands given here

Runtime Issues

One of the most common issues appears directly after installing OLAC Drives. The browser will be opened and the following screen appears:



There are several reasons that will inevitably lead to this problem:

- you did not power on the drive before starting OBLAC Drives
- you selected the wrong network port while configuring the VMware settings (check out our Setup Guide)

Other Issues

If the application does not behave properly in any other way than described above, try the following:

- Refresh the OBLAC Drives web page
- Make sure no native EtherCAT master is running on the host machine or in the same network
- Make sure that none of the Synapticon tool components (OBLAC Drives, motion master bridge or motion-master) are already running on the host machine as a native installation
- If your laptop is connected to a docking station, undock and restart it, then try to run OBLAC Drives again
- Other installed or running software on the PC is occupying the Ethernet port.

  If possible, stop or uninstall this software or try it with a fresh PC.
- Not all network adapters are able to perform the real-time requirements for OBLAC Drives. In our experience, network cards inside laptops and computers work most reliably. Docking stations (if not connected via USB) are also often reliable.

  A network connection via USB is difficult to predict. Some adapters work very well, others not at all. If USB is the only option, you may have to try several adapters or even a different PC. This is most likely the cause if you can establish a connection to a drive, but the connection is interrupted frequently.
- Limited performance of the host PC. If the general performance of the PC is too weak (less than 8 GB RAM, low frequency) or too many other software is running on the PC, there is not enough power left for the virtual machine.

  Stop all other software that is running or try with a different, more powerful PC

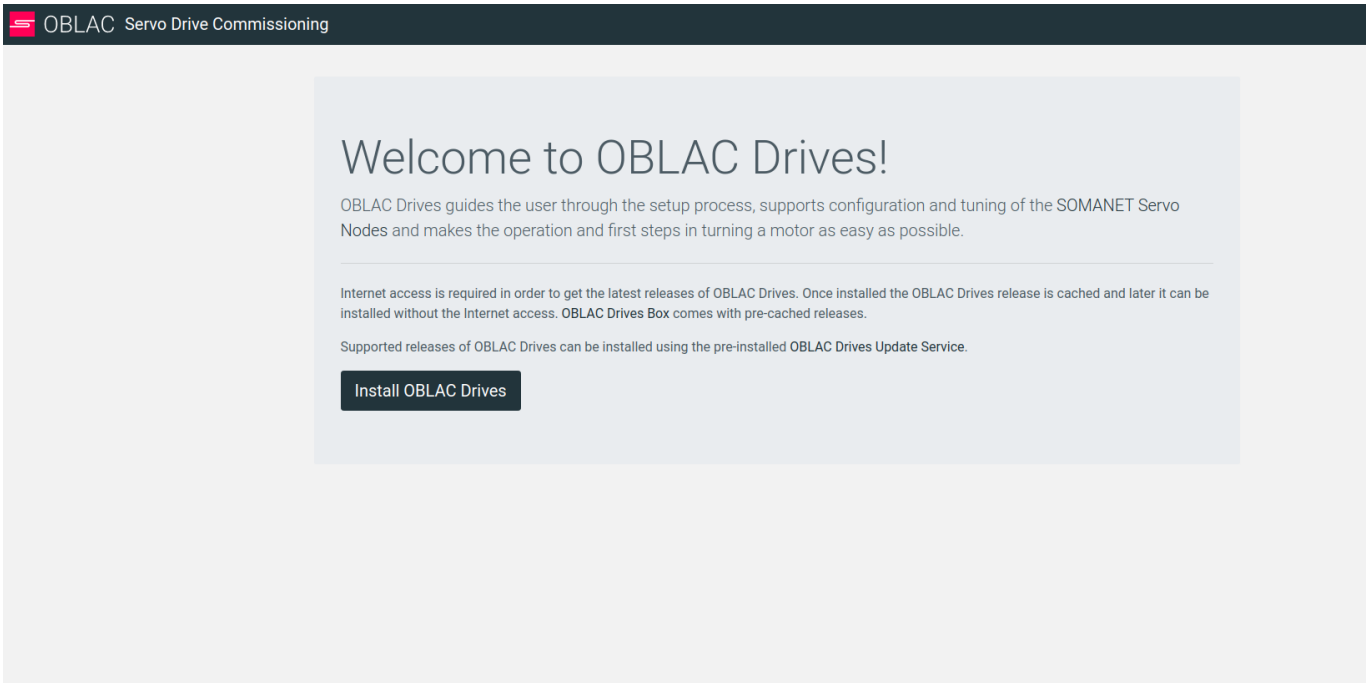
Note:

If you couldn't find a solution to your problem, please contact us at support@synapticon.com.

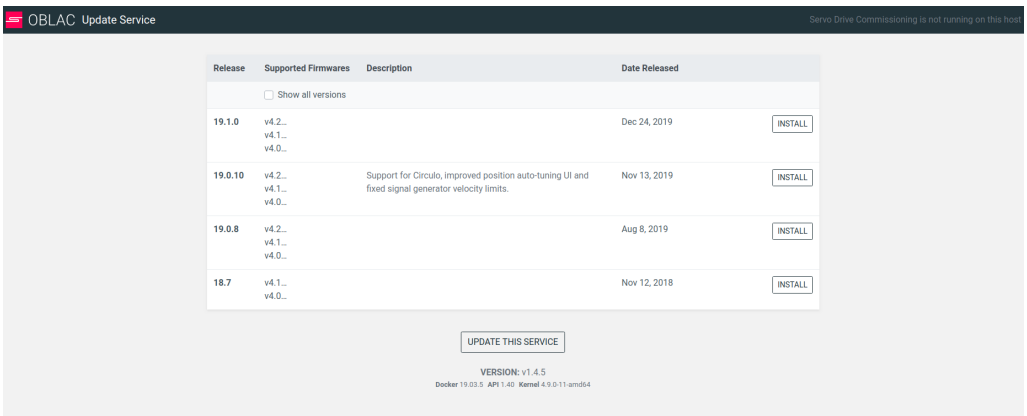
Using OBLAC Drives

Installing the latest version

After you have successfully commissioned OBLAC Box or powered on the OBLAC Drives image you will be asked to install the latest version:



After clicking "Install OBLAC Drives" you will be forwarded to the OBLAC Update Service and see a list of all current releases.

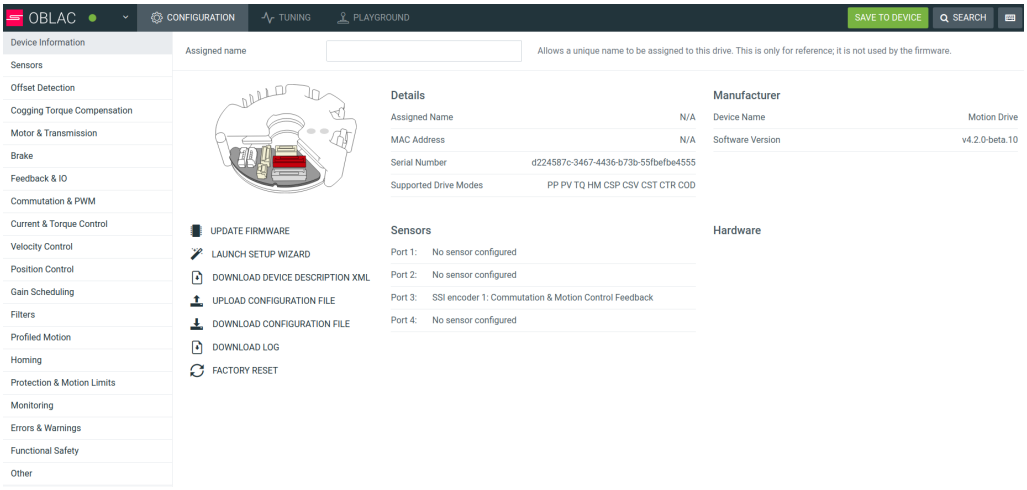


Use this list also to select between different versions of OBLAC Drives.

> Note:
>
> Please ensure that the release you want to install supports the firmware that you want to run on your drive.

See details for Updating/downgrading OBLAC Drives

User Interface Overview



Save to device

This operation will save all parameters to the config.csv file. When the device is power-cycled it will load this file.

> Note:
>
> When a parameter is entered, this change will be lost after power-cycling the device unless it is saved to the device manually.

Update Firmware

When this button is clicked, a box appears listing the most recent firmwares available for your hardware.

The list is being filtered by the device hardware description and the running release of OBLAC Drives.

Launch Setup Wizard

This intuitive tool helps you enter the specifications of attached motors, sensors, gear and brake.

Details can be found in the section Set up your servo drive.

Download Device Description XML

The Device description is an XML file that contains all the objects used and their properties.

For EtherCAT devices this is ESI (EtherCAT Slave Information).

Load Configuration File

This action let's you select a previously saved configuration file from your computer and load it onto drive effectively replacing the current parameter values with the ones in the file.

Save Configuration File

Download your configuration if you want to use it on other drives.

Download log

The log contains the state changes and all occurring errors and warnings.

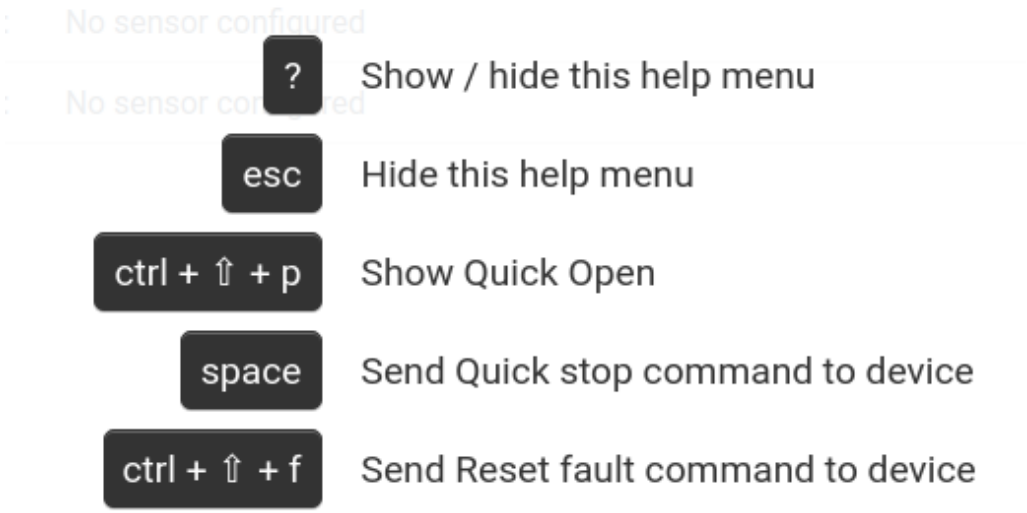
Factory Reset

When serious issues occur, you can set the system back to Factory settings.

> Attention:
>
> This action is irreversible. It will remove all files from a drive and install an empty firmware.

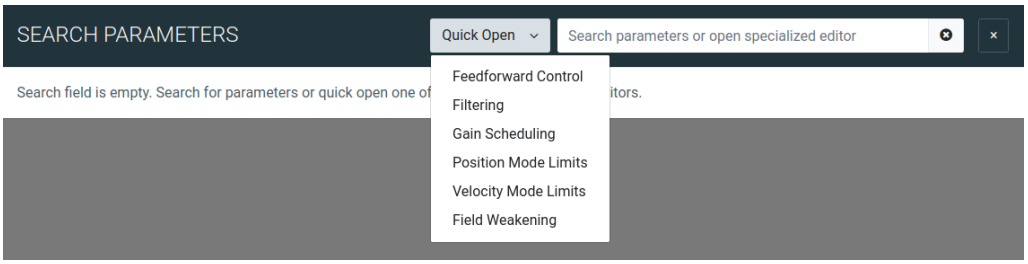
Using keyboard shortcuts

Press the ?-key to get an overview of the shortcuts available.



Quick Open

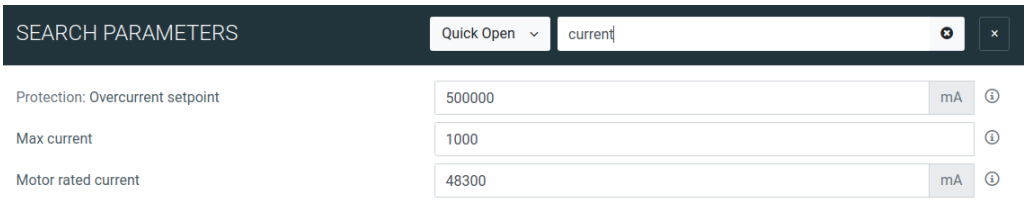
Clicking on the search field accesses the Quick Open feature. You can also access it by entering the shortcut `Ctrl-F`.



In the Quick Open field, any keyword typed in will reveal the parameter that matches best.

The search input can search parameters by:

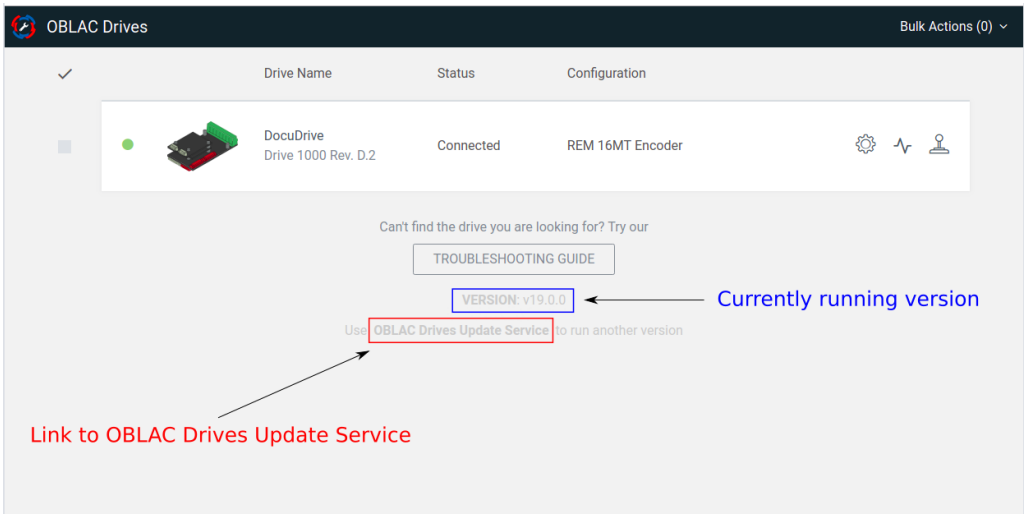
- name
- index in format 0x1000
- specialized editors prefixed with # for example `#filtering`.



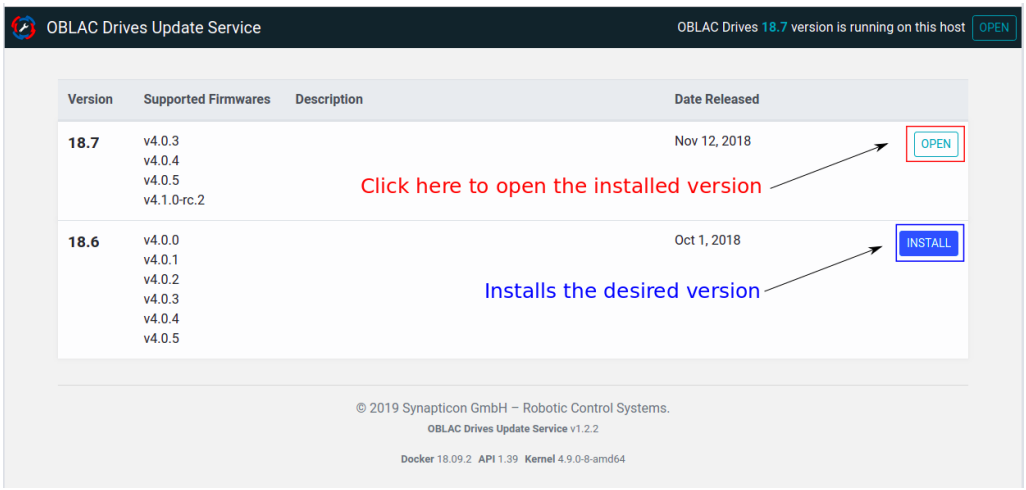
You can edit the value directly in the Quick Open input box or view the parameter documentation by clicking on the (i) button.

Update/downgrade OBLAC Drives

The pre-installed version of OBLAC Drives may need to be updated. You can also downgrade to a previous version if your drives requires it.



Click on the Link to open the [OBLAC Drives Update Service]

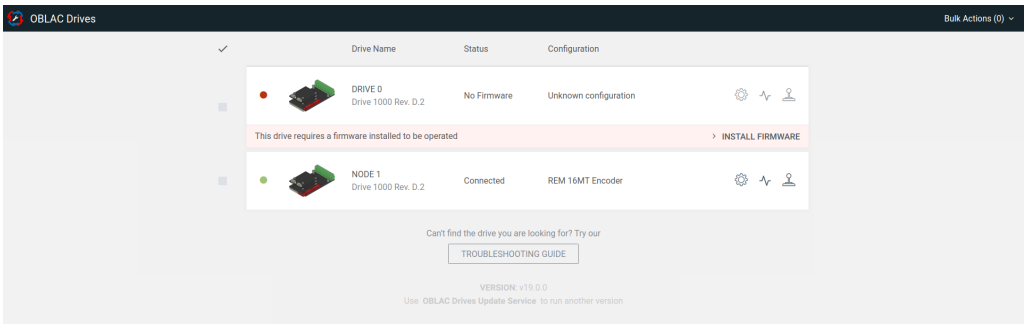


1. From the list of releases, click “INSTALL” on the release of your choice.
2. After installation, click “OPEN” to return to OBLAC Drives.

> Note:
>
> When you have downgraded to a version older than 19.0.0 you wont find the Link anymore.
>
> To access OBLAC Drives Update Service please manually enter port 64000 in your URL (e.g 192.168.0.1:64000)

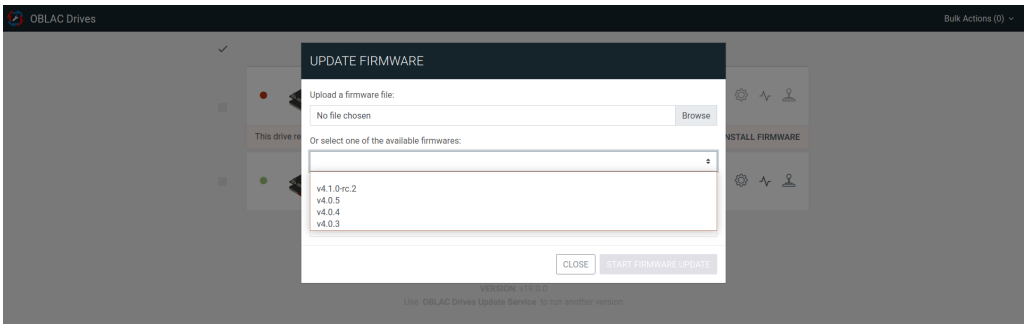
Set up your servo drive

Once OBLAC Drives has been installed you can open a browser window using the IP address shown in your virtual machine and you will find all the connected drives. In our example, the connected drive does not have firmware installed (which is the default state after delivery of our drives).

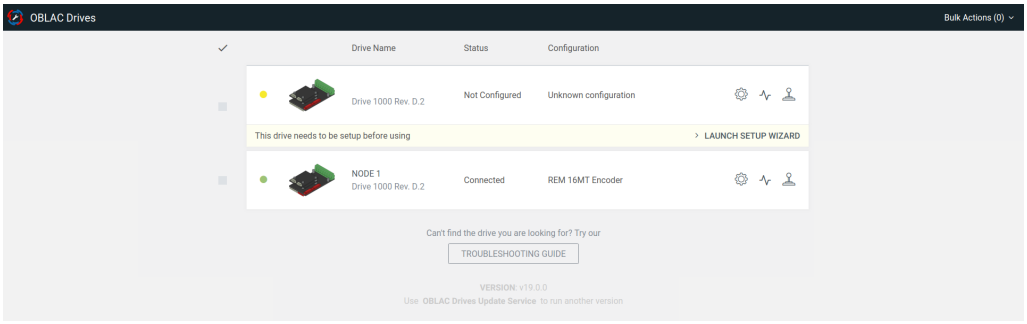


When clicking on the “Install firmware” link, a dialog box is opened in which you can choose the latest version of the firmware at the top of the list. Once you clicked on the “Update” button, please wait until a notification pops up telling you that the installation is done.

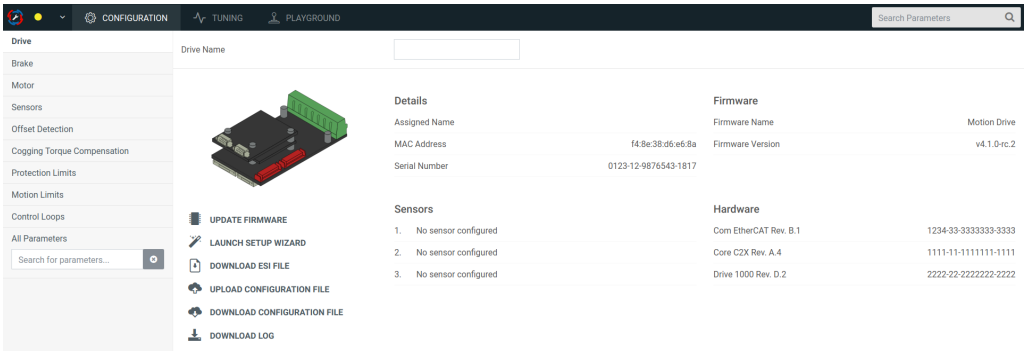
> Note:
>
> Please make sure to select a firmware that fits your version of the Drive module.



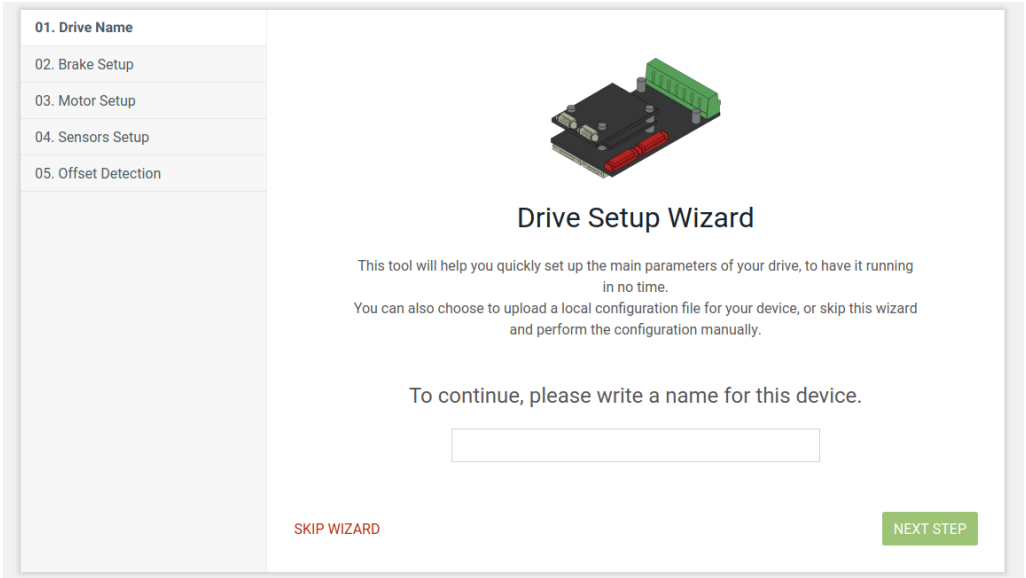
After the firmware is installed, OBLAC Drives will restart the tools, this will take a few moments. Once the connected drives are initialized again, OBLAC Drives will show a message that the drive isn’t configured yet.



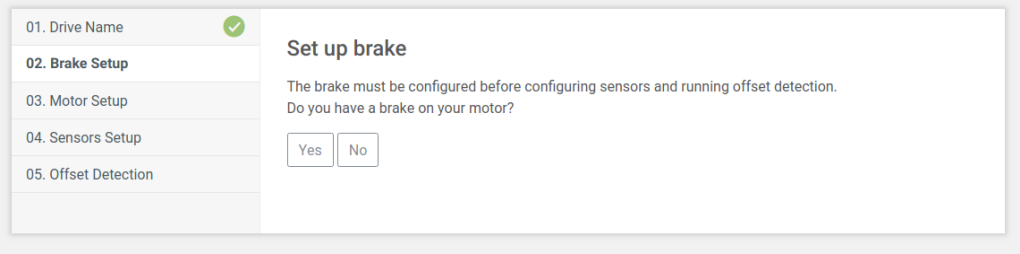
> Note:
>
> If you want to update your firmware, please click on the drive icon and select **Install Firmware**



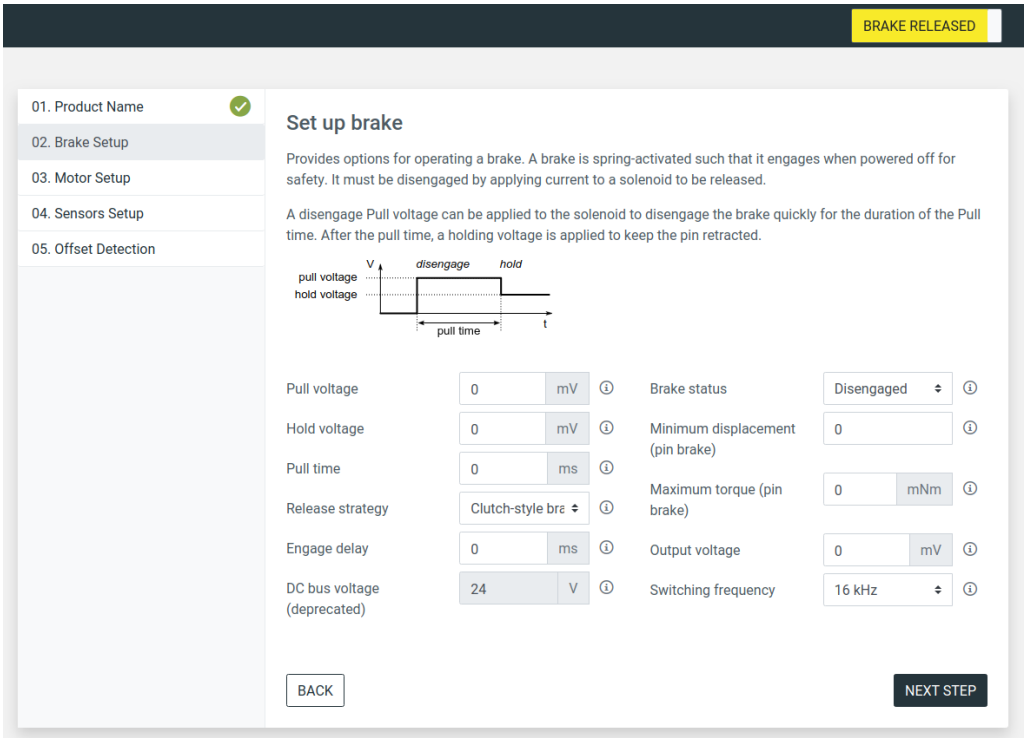
Click on the “Launch Setup Wizard” link to open the setup wizard. It will first ask you to give a name to your drive.



First you need to determine if you have an attached brake.



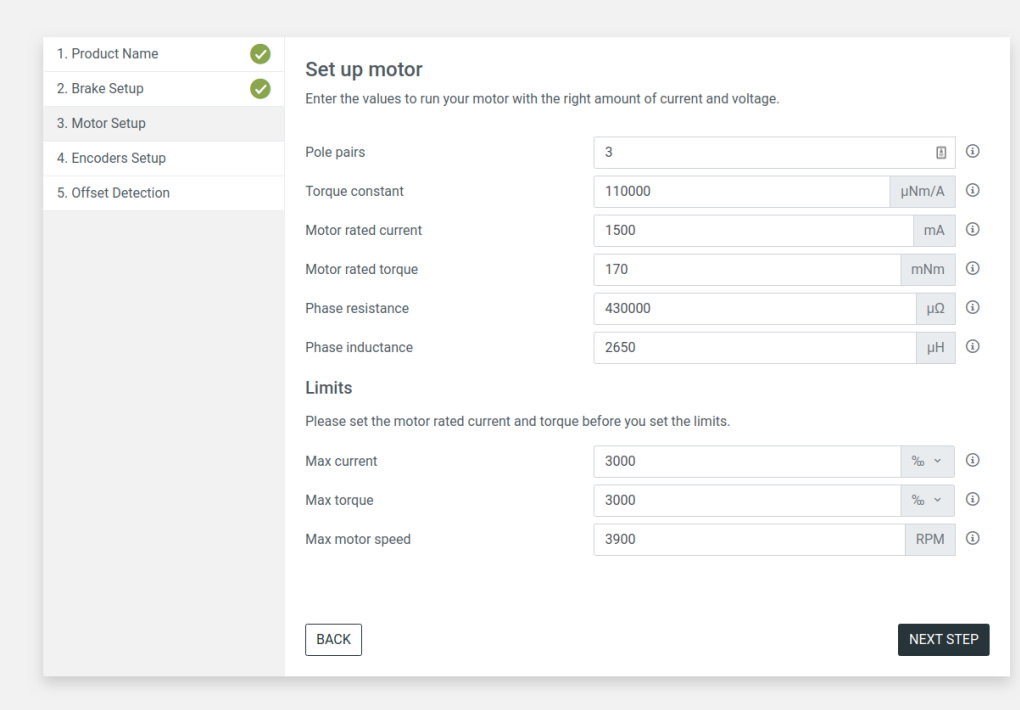
If you have a brake, insert the parameters.



The release-strategy will be set to clutch-style brake by default.

> Note:
>
> When you click the button to release the brake, it will apply a pull voltage to the brake but it will not perform the behavior for shaking free as described in the brake release feature. So if you have a pin-style brake, please ensure that no heavy load is attached to the motor.

After clicking on “Next” you will be asked for details of the connected motor. Add the parameters of your motor according to the data sheet.

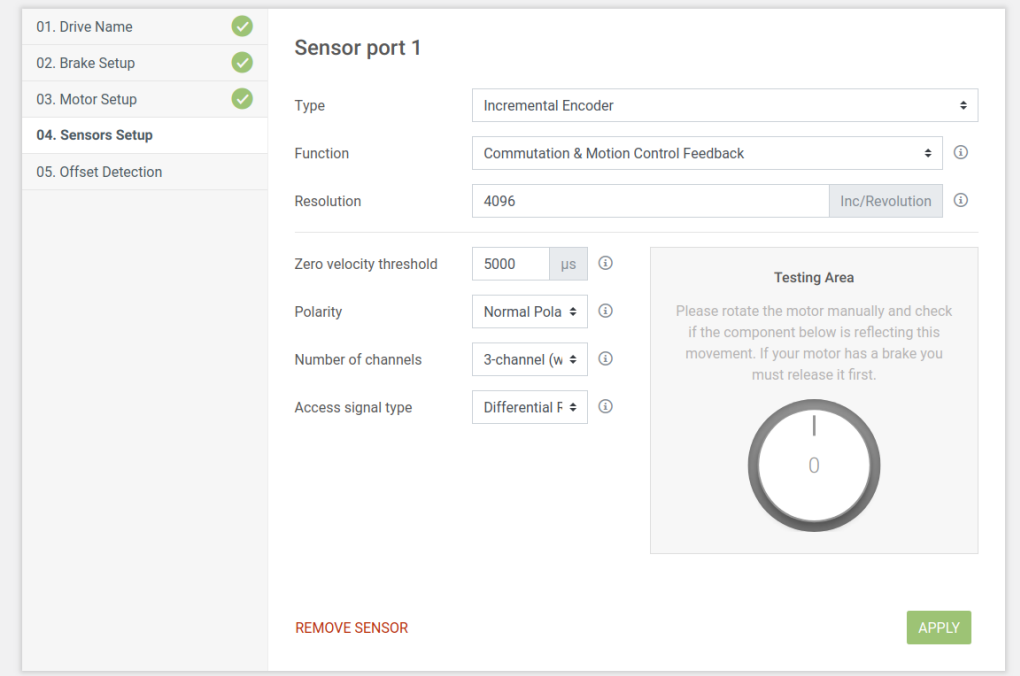


Then it shows a sensor configuration page, where you can add, remove and configure your connected encoders.

> Attention:
>
> It is of vital importance to have **exactly one** encoder configured for Commutation so that the motor shaft's position is known to the commutation algorithm. In case your setup requires it, you can use a different encoder for Position or Velocity. In this case, the demanded and actual velocities will be scaled to the same reference frame of the position.

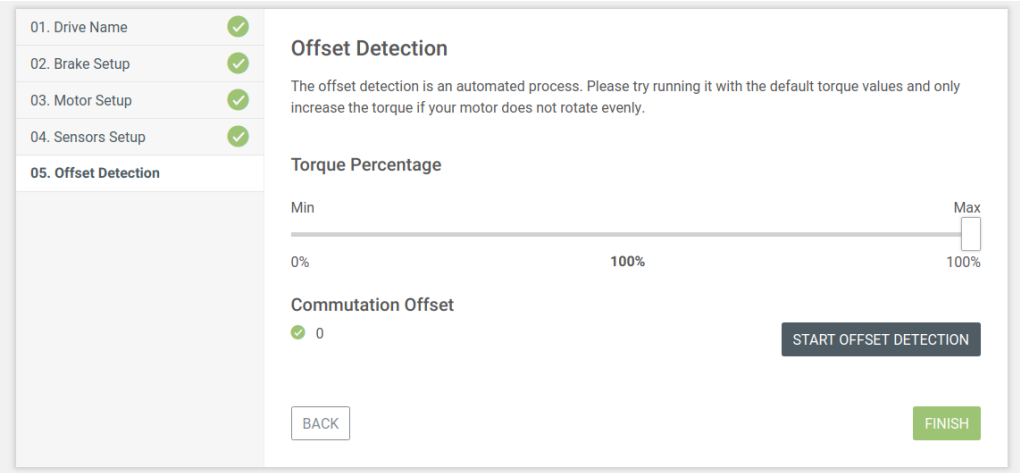
To configure an encoder, click onto "details" and set the parameters as you wish. If you want to use this encoder as a motion control sensor, you will be able to test if it is properly configured in the testing area.

Details can be found in encoder configuration.

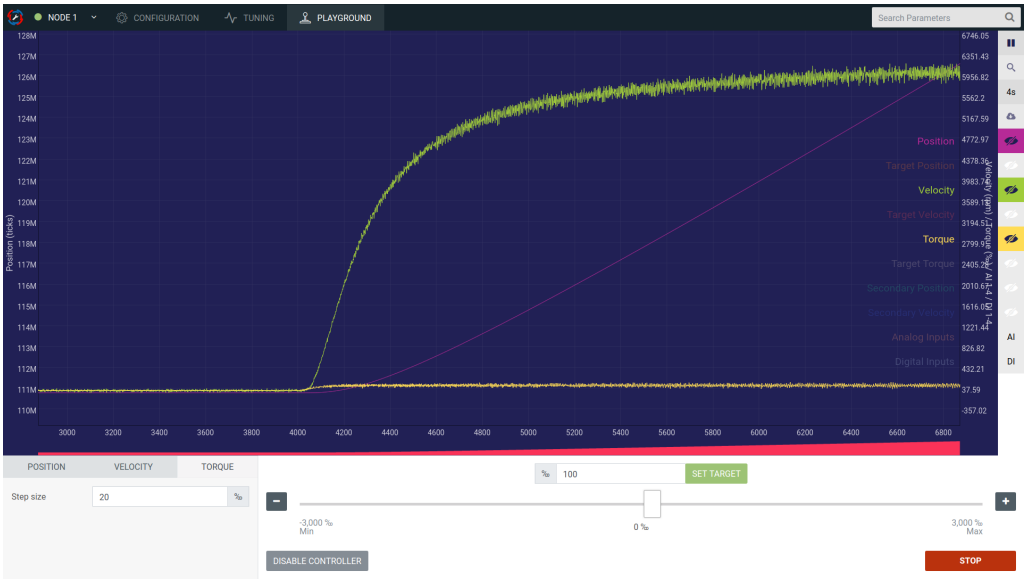


The last step of the setup wizard will guide you through the offset detection. The procedure starts with a value of 100% which should yield good results.

> Note:
>
> When using a power supply that can't provide the rated motor phase currents, set the slider to a lower value. This may be indicated by the error "Undervoltage".



After finishing the wizard, OBLAC Drives will open the "Playground", where you can test the basic functionality of your new SOMANET drive setup.



Test a drive at the playground

Prerequisites

- The servo drive is connected to power supply, motor, position sensor and to a computer.
- OBLAC Drives is installed, running and established a connection to the servo drive.
- The latest firmware is installed on the drive.
- All basic configuration parameters of the motor and sensor are copied from their data sheets to the drive configuration.
- The position sensor provides plausible data, the commutation offset was found.
- Ideally, the Autotuning was performed successfully.

Applying torque to a motor

If all the prerequisites ar met, the motor can be tested.

1. Go to OBLAC Drives.
2. Switch to the Playground section from the top menu.
3. Select “TORQUE CONTROL”.
4. Use the slider or the buttons to enter a reference torque.

> Important:
>
> - Applying a constant torque means that the motor shaft will start moving and not stop

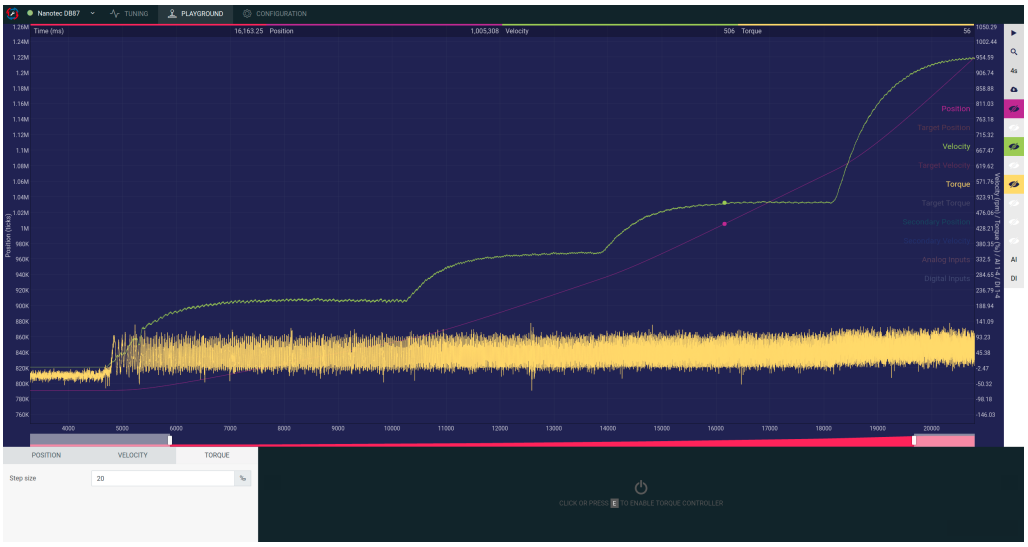
before you manually set it back to a lower value. If anything is attached to your motor shaft (e.g. robotic manipulator), make sure there are no mechanical limits to be exceeded or anything might be hit.

- Torque is given in the unit per thousands of rated torque.

So, if the motor's data sheet specifies T_rated = 0.5 Nm and you entered the corresponding 500 mNm into your configuration, setting the torque setpoint to 100 will lead to: 100 * 1/1000 * 0.5 Nm = 0.05 Nm

- Usually, a value between 50 and 300 should be sufficient to make any motor start rotating - regardless of its absolute size and rated torque

The motor should start rotating now and you should be able to observe the motion in OBLAC Drives.



Rotate a motor at defined velocities

One of the next steps can be testing the velocity control mode. The Playground allows to set and alternate the velocity setpoint. Before, the Velocity controller has to be tuned. Select the velocity increment size, enable the velocity controller and try out different velocity setpoints. Position, torque and other signals can be observed if needed.

You may also configure additional features such as:

- Velocity feedback signal filtering
- Cogging torque compensation

- Velocity offset for Feed-forward control
- Field weakening

Rotate a motor to defined positions

Common tasks in robotics involve moving a manipulator, vehicle or gripper to a predefined position. In order to do so, your SOMANET servo drive comes with an internal position controller that can be configured and adapted to your surrounding mechanical system automatically.

Enter a target position in the units of encoder increments (‘ticks’) and see how the motor moves there. Depending on your encoder resolution, quite large numbers are necessary to create visible motions. For example, with a 16-bit encoder 65,536 ticks represent one motor shaft revolution. If you even have a reducer (gearbox) attached to your motor shaft, you may have to change the position setpoint by several 100,000 ticks to create visible motion. However, we recommend to start with small steps and then slowly increase them.

Finally, you can check the performance of your position control loop by using one of the position setpoint test functions available in OBLAC Drives. Depending on what suits your application best, you can choose from:

- step
- sinusoidal
- trapezoidal

position reference signals.



Use the buttons on the right hand side of the window to see detailed plots of the tracking behavior (position setpoint and actual position) and other values like the torque. You can use the figures to

check if the overshoot, response times or bandwidth of your entire control loop suit your application.

If you encounter any problems, make sure they are sorted out before you continue. Reviewing the above-mentioned list of prerequisites should be your first step. You can also have a look at our troubleshooting site.

Motion Control Tuning

Controller Tuning is an essential task in setting up a Motion Control system. It ensures that the motor behaves exactly as you expect it. Our Auto-Tuning feature will provide good results for most applications but you can still manually fine-tune your system.

Autotuning Overview of the Autotuning feature

System Identification This feature allows to generate a precise model of the mechanical system with one click

Position Autotuning Automatically generates gain values for the cascaded position controller

Velocity Autotuning Automatically generates gain values for the velocity controller

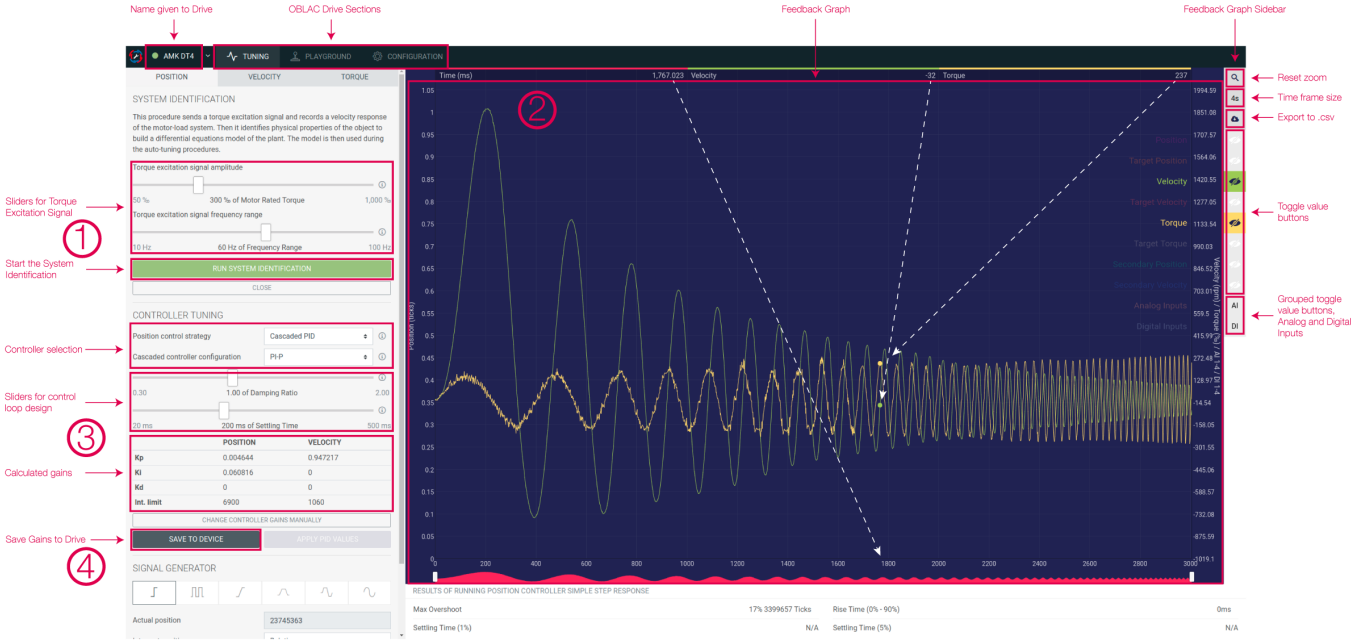
Manual Tuning Description of steps required for manually tuning the controller gains

Manual Tuning of the current controller The generated torque is directly proportional to the current.

Manual Tuning of the velocity control loop Instructions for manually tuning the proportional and integral gains of the velocity control loop if the drive should be operated in velocity control mode.

Manual Tuning of the position control loop Instructions for manually tuning the cascaded PID controller.

Overview of the Tools



1. Start tuning by identifying the mechanical properties with our System Identification feature
2. Use the feedback graph to determine the best values
3. Finetune the controller gains and design the optimal control structure
4. Save the values to the drives

Auto-Tuning

The manual describes how to tune the cascaded position controller using the Auto-tuning feature.

Before conducting any of the tuning activities please check if motor is functional in torque mode and sensor data is transmitted normally. For high-quality motor control tasks, a precise and reliable sensor should be used. Quality of the motor position control may be insufficient with low resolution sensors, especially if a Hall sensor is used.

The current version of the auto-tuning feature allows user to tune a BLDC motor drive. A load can be attached to the motor. However, if the motor position has very strict limits this feature is not recommended, because the motor may make a few full turns.

Concept of Auto-Tuning

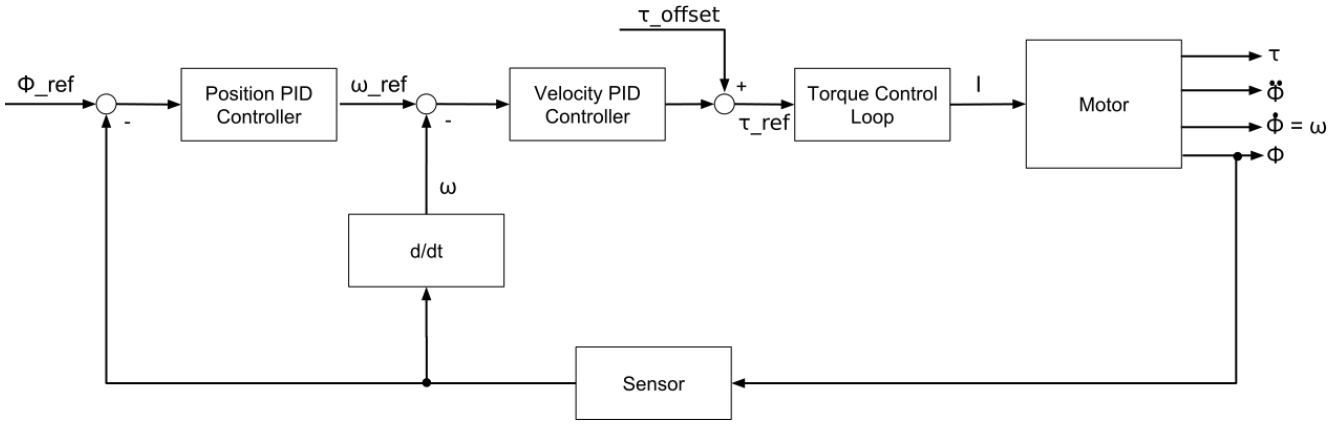
Auto-tuning is performed in two steps:

- System identification Identifying the physical properties of the object to build a mathematical model of the target system.
- Depending on the application, either position or velocity Auto-tuning can be performed:

- Position auto-tuning Using the model to calculate controller gains that result in a specified position control-loop performance.
- Velocity auto-tuning Tuning the velocity loop with Velocity Auto-Tuning for Cyclic Synchronous Velocity mode (CSV) or Profile Velocity Mode (from firmware 4.2 on).

Goal of Position Auto-tuning

The Position Auto-tuning process calculates the gains for the cascaded controller automatically.

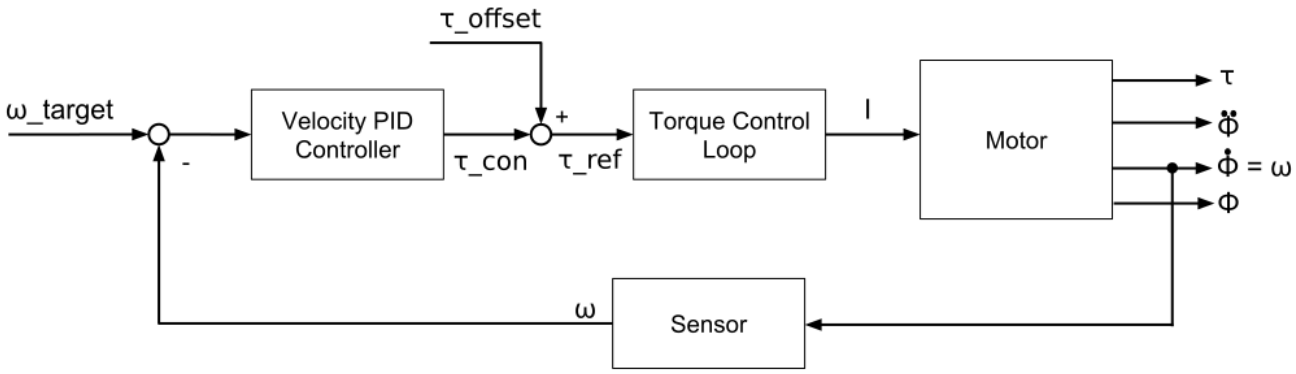


The core of the cascaded controller is linear. However, there are multiple nonlinear additions: saturation, feedforward terms and integrator clamping.

> Note:
>
> In our experience, the cascaded PID controller structure is one of the most effective for motor control applications. For details see PID Controllers: Theory, Design, and Tuning

Goal of Velocity Auto-tuning

The Velocity Auto-tuning process calculates the gains for the velocity controller automatically.



System Identification

An open-loop system identification extracts a mathematical model of an electromechanical system from a frequency response. The obtained model is then used during the auto-tuning procedure.

Prerequisites

Before conducting system identification make sure that:

1. The motor is connected and functional.
2. The system is firmly fixed.
3. The motor shaft can rotate.
4. The torque control loop is functional and the commutation offset is found.
5. Position and velocity signals are present and output sensible data.
6. Applying constant positive torque results in positive velocity and vice versa.

> Note:
>
> Using a Chopper Board is strongly recommended to prevent hardware damage

Limitations

1. The procedure should be used carefully if the system has strict mechanical limitations. Torque amplitude should be decreased until safe identification is possible.
2. If the system is highly non-linear, the identified model might differ from the reality.
3. If motion and vibration are too high, the algorithm can't be used.

4. The accurate frequency response of the system can be obtained only for frequencies up to 100 Hz due to communication limitations.
5. If the system has very high static friction, small amplitude torque signal will overestimate viscous friction and inertia.

How to use

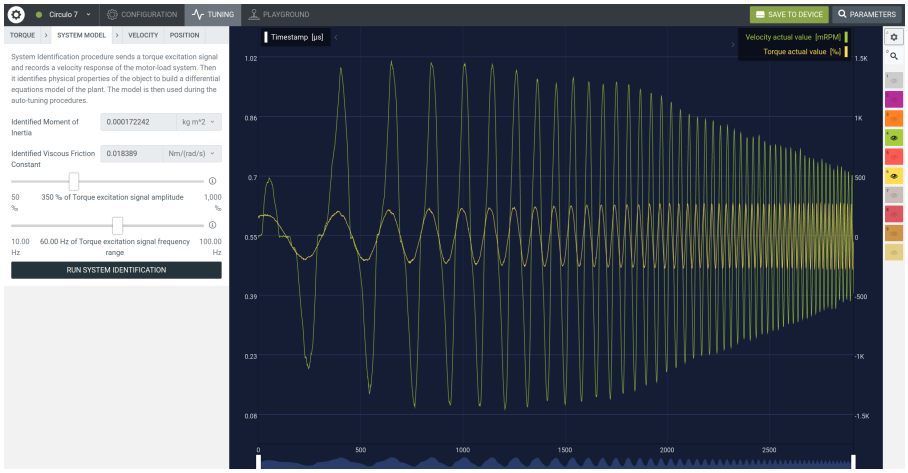
> Note:
>
> To reset to default values simply click on the Velocity or Position tab.

1. For general applications the default settings can be used. Press “Run system-identification”.
2. Note that the amplitude of the torque test signal changes linearly from 50% to 100% of the displayed value.
3. Make sure that the system moves during the experiment. Velocity and torque signals on the screen are nonzero. If the system did not move, the torque amplitude should be increased. A higher motion amplitude is recommended to ensure the system is in the viscous friction region.
4. Wait until the procedure is finished.
5. Increasing the identification signal frequency range will result in detecting resonances of the system. However, inertia estimation will be less precise due to overfitting.
6. There are 2 sliders that allow users to change the frequency response test signal:

Torque excitation signal amplitude - specifies a max percentage of the rated torque applied during the identification. Torque amplitude increases linearly from 50% to 100% of the slider value to ensure less motion.
How to use this slider:

- Adjust the slider from smaller to bigger values so the system moves in a safe range and vibrations are still acceptable. The resulting velocity should be nonzero and have the same sign as the torque at the beginning of the experiment.
- Low values result in smaller vibration amplitude, less stress on the mechanical structure but low signal to noise ratio may result in an imprecise model. Additionally, if static friction is present and the system moves slowly, the algorithm will overestimate the load inertia and the friction constant. An example of low

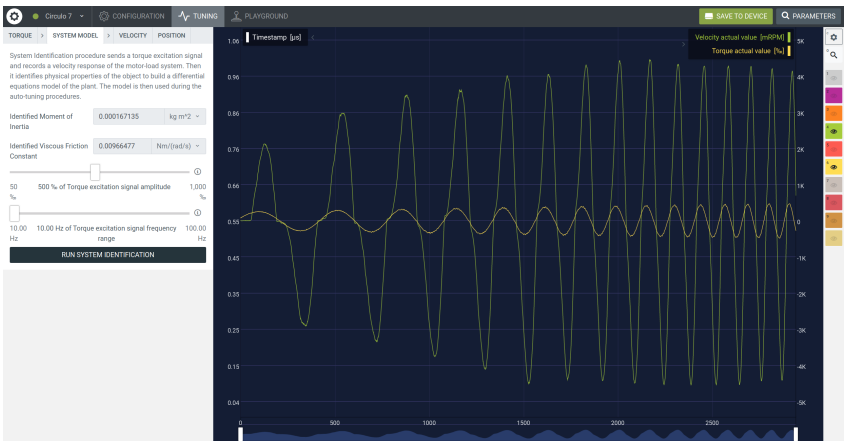
amplitude can be seen in the image below:



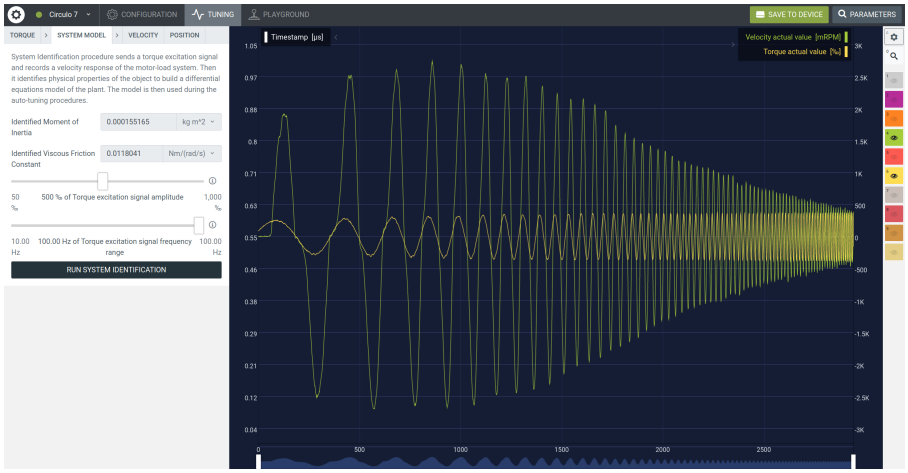
- High values result in a better signal to noise ratio and allow to overcome static friction if the system doesn't move with default input. However, oscillations may have a bigger amplitude and generate noise in the torque controller.

Torque excitation signal frequency range - defines the frequency range of the injected signal: from 2 Hz to the specified value.
How to use this slider:

- It is recommended to use the default value. Adjust the slider if a known resonance should be included into the model.
- Low values will result in a more precise identification procedure and more stress on the mechanical structure.



- High values will result in less stress and the identification procedure will be less precise due to the EtherCAT communication delay. As a result, higher frequencies are included into the model. An example of high frequency range can be seen in the image below:



When the System Identification was successfully executed the dialog collapses to a button “Rerun System Identification”. A set of default controller gains is calculated and the options for position autotuning are displayed instead.

> Note:
>
> If the electromechanical system was changed the identification should be repeated.

Now you can calculate the controller gains with Position Autotuning for a specified position control-loop performance.

Or tune the velocity loop with Velocity Autotuning for Cyclic Synchronous Velocity mode.

F.A.Q.

Can I define **other velocity/torque units** in the custom firmware?

- The system identification is possible only with default units.

Do I have to repeat system identification after **attaching a load to the system**.

- It is recommended to repeat the procedure.

What if my encoder has a **low resolution**?

- The velocity signal used in system identification might have multiple zeros that may result in an incorrect model.

The system goes to **overcurrent** or the **software resets** during the system identification.

- Check the configuration parameters and compare them to the official motor and sensor data sheet.

- Ensure that the torque controller works correctly.
- Lower the torque excitation signal amplitude.

Position control loop

The general purpose of the position control loop auto-tuning is to design position controller gains based on user specifications and the identified model of the set-up. The procedure calculates gains for the cascaded controller configurations: PI-P, P-PI.

Prerequisites

Before conducting position auto-tuning check the following:

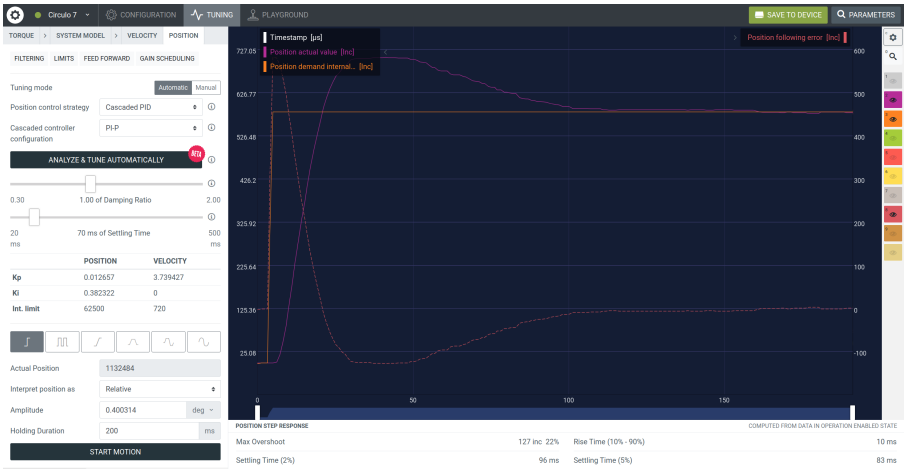
1. The System Identification has been executed successfully.
2. The obtained model corresponds to the current configuration of the system. Otherwise the identification procedure should be repeated.
3. If different sensors are used for position and velocity control and there is a gearbox in between, the gear ratio should be specified.
4. Make sure that the sensors are mounted properly and calibrated.

Limitations

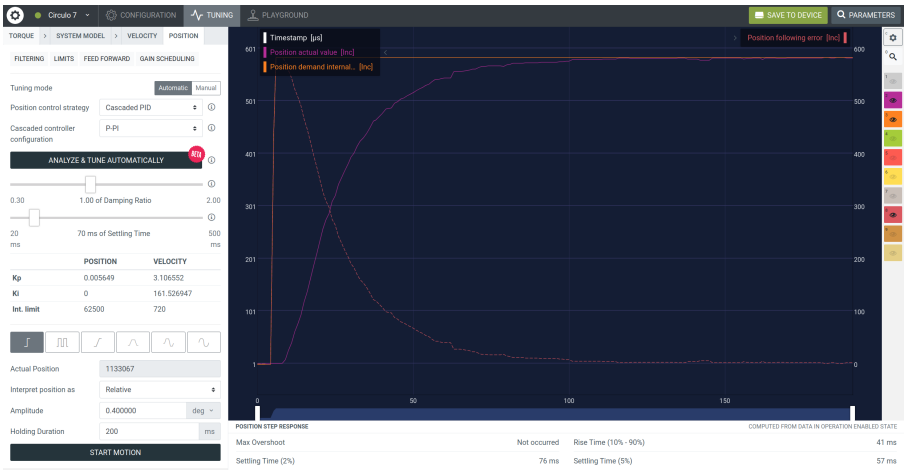
1. The procedure uses a linear mathematical model of the system. If there are strong non-linearities there will be a difference between the actual and the requested performance.
2. The auto-tuning procedures tunes only linear controllers. Non-linear control parts should be tuned manually for better performance.
3. Hall sensors should not be used for precise position control.

How to use

1. Select “Controller Configuration” from the menu. Currently, two options are available that have different behavior:
    1. PI-P configuration (default) focuses on trajectory following. Generally it has a faster response, zero steady state error for constant and ramp reference inputs. The control loop bandwidth is higher than the P-PI configuration. However, it tends to overshoot:

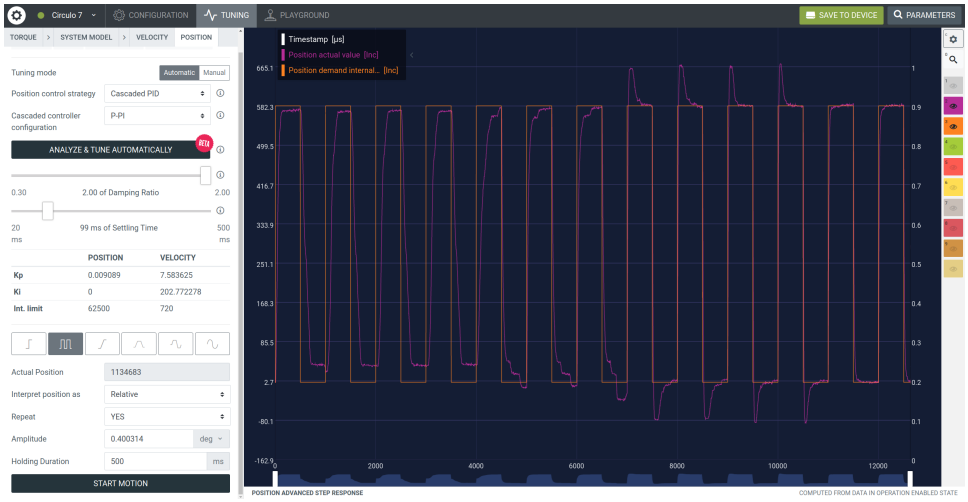


2. P-PI configuration focuses on soft damped step response. It has a slower response and less precise tracking, however it is usually possible to fully avoid overshoot. Additionally, it is more sensitive to noise in the velocity signal:



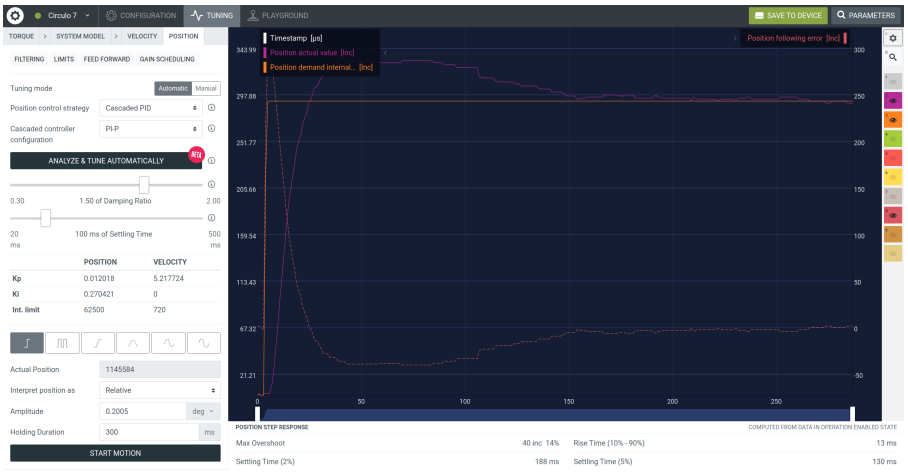
2. The controller gains are updated automatically every time the sliders are dragged and released. The gains calculation has an optimization loop inside and may take a few seconds.

3. After the initial tuning is done, it is possible to tune the control loop more precisely by changing the gains while executing a test signal. There are currently 4 repeatable test signals: square wave, ramp, bidirectional ramp and sine. It is possible to drag sliders and manually change gains during the reference signal following:

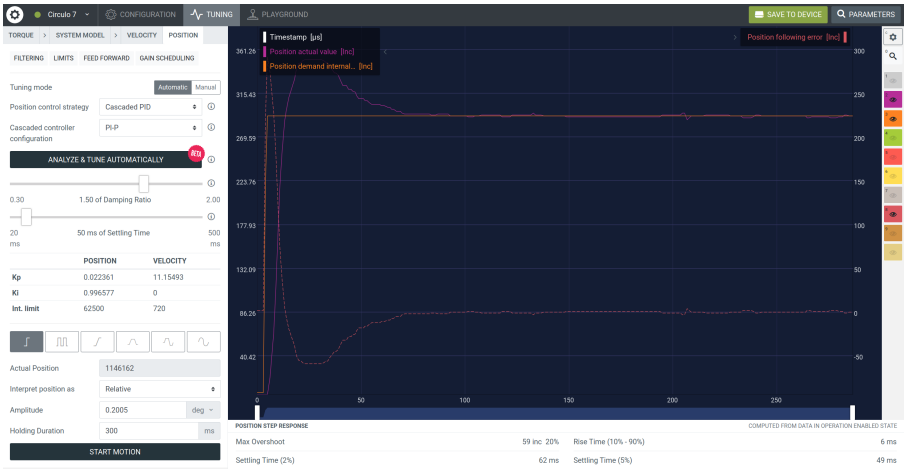


4. There are 2 sliders that allow users to specify the demanded performance of the control loop:

    1. **Settling time** is the time [in ms] needed to settle the step response within 2% of the reference value. The auto-tuning procedure will try to find gains to ensure the specified settling time given the identified model.
    Settling time of 100 ms:



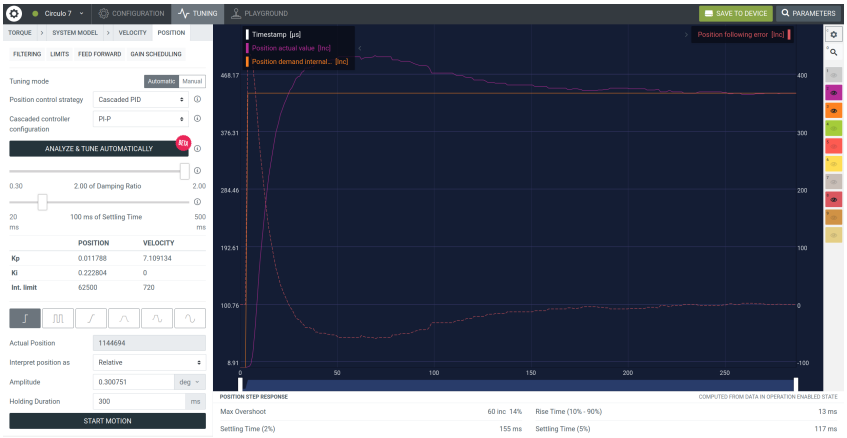
    Settling time of 50 ms:



2. Damping ratio is a parameter that reduces oscillations in an oscillating system. In a cascaded control loop, this parameter indirectly influences overshoot and rise time. Practically, the damping ratio effect depends on the controller type.

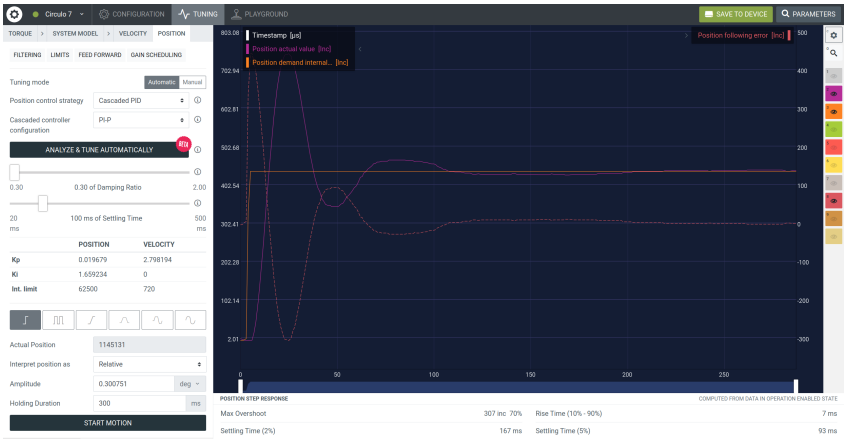
    - Values > 1 can be used for removing overshoot but at cost of slower settling time, it may also lead to undershoot and non-zero steady state error.

      Damping ratio is maximal:



    - Values < 1 can be used to achieve a sharper response and a better bandwidth at a cost of overshoot:

      Damping ratio is minimal:



F.A.Q.

Do I need to manually adjust controller gains after the auto-tuning?

- Usually, sliders provide enough flexibility to achieve demanded performance. Mismatches between requested and actual step response can be compensated by entering a faster settling time.

Can I define other position/velocity/torque units in the custom firmware?

- The auto-tuning will provide gains only for default units.

What if my encoder has low resolution?

- This will definitely result in worse performance compared to higher resolution encoders. Additionally, the velocity signal used in system identification might have multiple zeros that may result in incorrect model.

Do I have to repeat system identification and tuning after I attached a load to the system?

- It is better to repeat both procedures. If it is not possible to repeat the system identification, auto-tuning still can be used. But the resulting settling time will differ from the requested.

What if the system is unstable after auto-tuning?

- Try to input a slower settling time.
- Check the motor and sensor configuration parameters.
- Check the velocity signal in the playground in the torque control mode. Ensure that it shows correct data during manual rotation of the shaft.
- Soften filters in position/velocity control loops. They might introduce delay.

What if the test signals trigger overcurrent/undervoltage?

- Reduce the test signal amplitude or frequency.
- Increase the settling time.
- Check the configuration of the motor and sensor.
- Make sure that correct power supply is used.

Is there a PID controller auto-tuning?

- A simple PID is usually not used for position control.

How do filters influence the auto-tuning?

- Using a velocity filter results in a phase delay based on the filter order and its cutoff frequency. It is recommended to perform a system identification and tuning after filters were configured. Be careful, filters severely reduce stability due to phase delay.

Velocity control loop

> Note:
>
> This feature is available since OBLAC Drives 19.0

The general purpose of the velocity control loop auto-tuning is to design controller gains based on user specifications and the identified model of the set-up. The procedure calculates gains for the PI velocity controller.

Prerequisites

Before conducting velocity auto-tuning check the following:

1. The System Identification has been executed successfully.
2. The obtained model corresponds to the current configuration of the system. Otherwise the identification procedure should be repeated.
3. If there is a gearbox in between, the gear ratio should be specified.
4. Make sure that the sensors are mounted properly and calibrated.

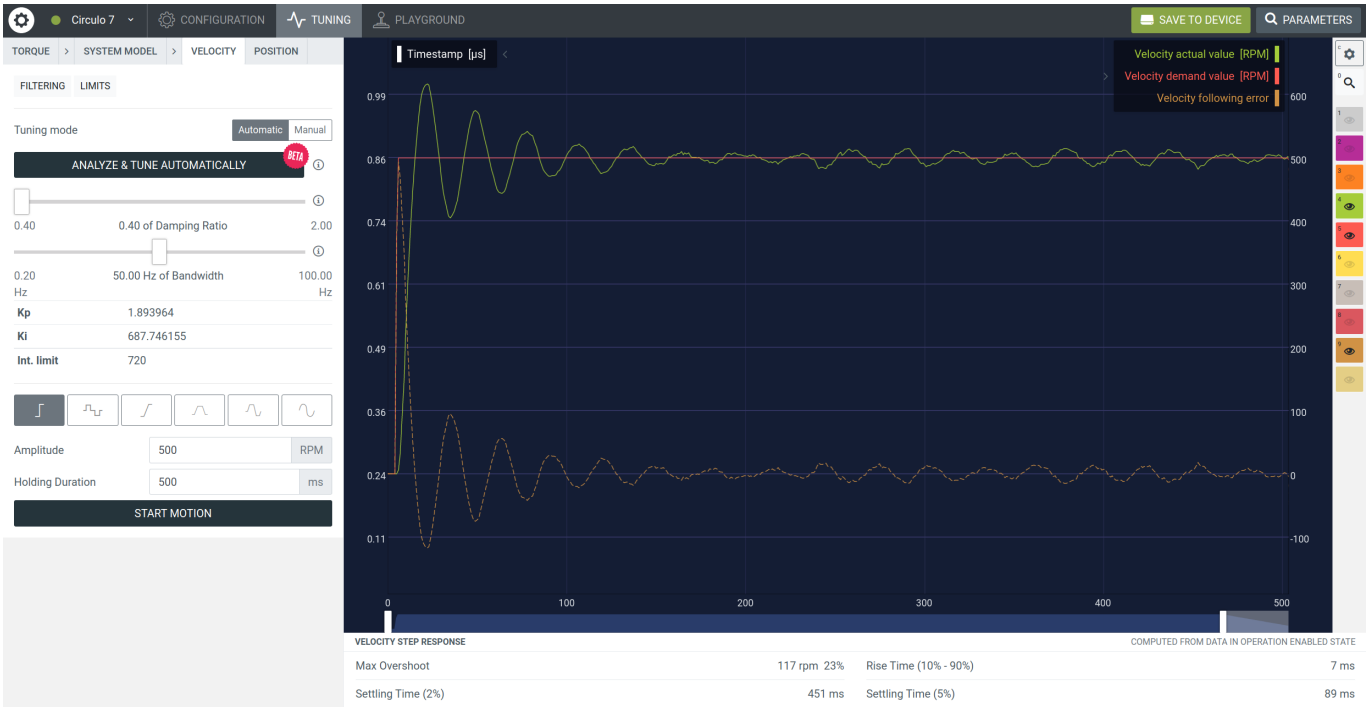
Limitations

- The procedure uses a linear mathematical model of the system. If there are strong non-linearities there will be a difference between the actual and the requested performance.
- **Max bandwidth** of 100 Hz is specified because of 1 kHz EtherCAT frequency.

How to use

1. The controller gains are updated automatically every time the sliders are dragged and released.
2. There are 2 sliders that allow users to specify the demanded performance of the control loop:

- **Damping ratio** is a parameter of an oscillatory system that reduces oscillations. In the velocity control loop, this parameter controls overshoot and rise time. The damping ratio can be divided into 2 regions:
  Underdamped < 1: overshoot is allowed, results in a sharper response.Example of a velocity control loop step response with a damping ratio of 0.4:

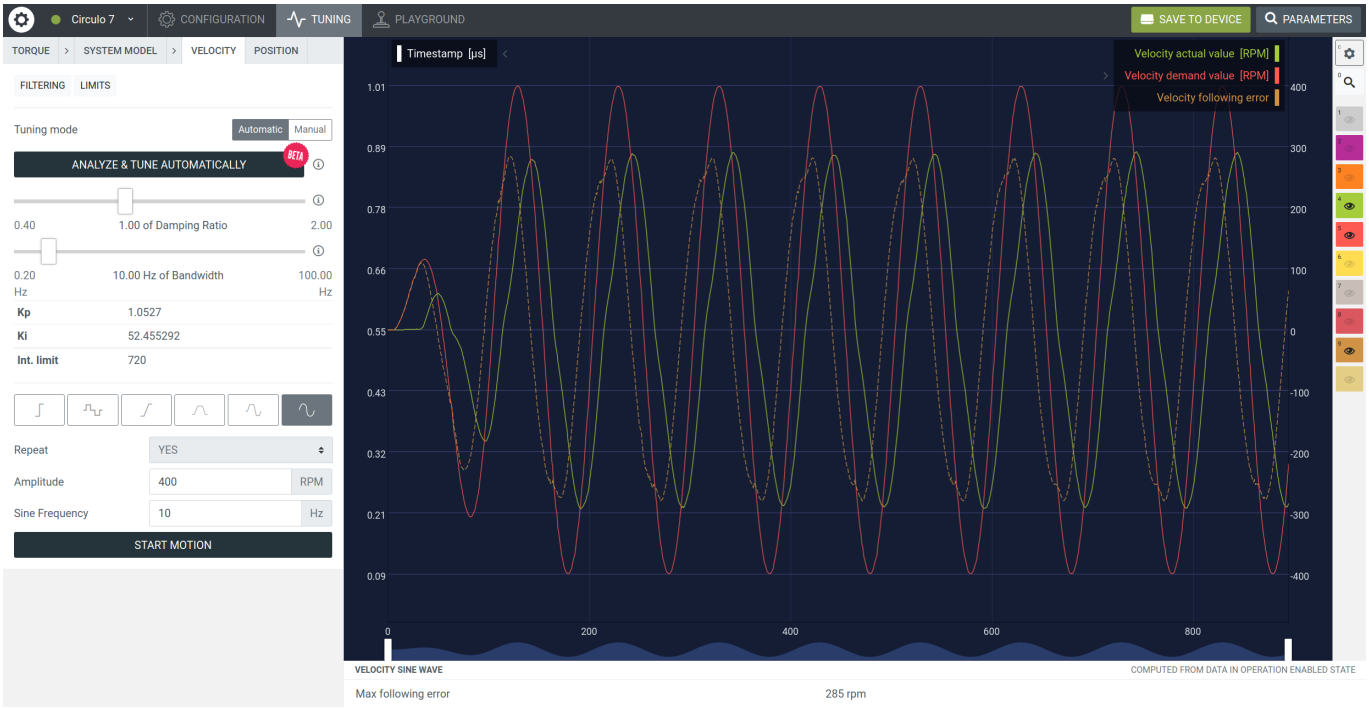


Overdamped > 1: no overshoot, results in a less sharp response.Example of a velocity control loop step response with a damping ratio of 2:

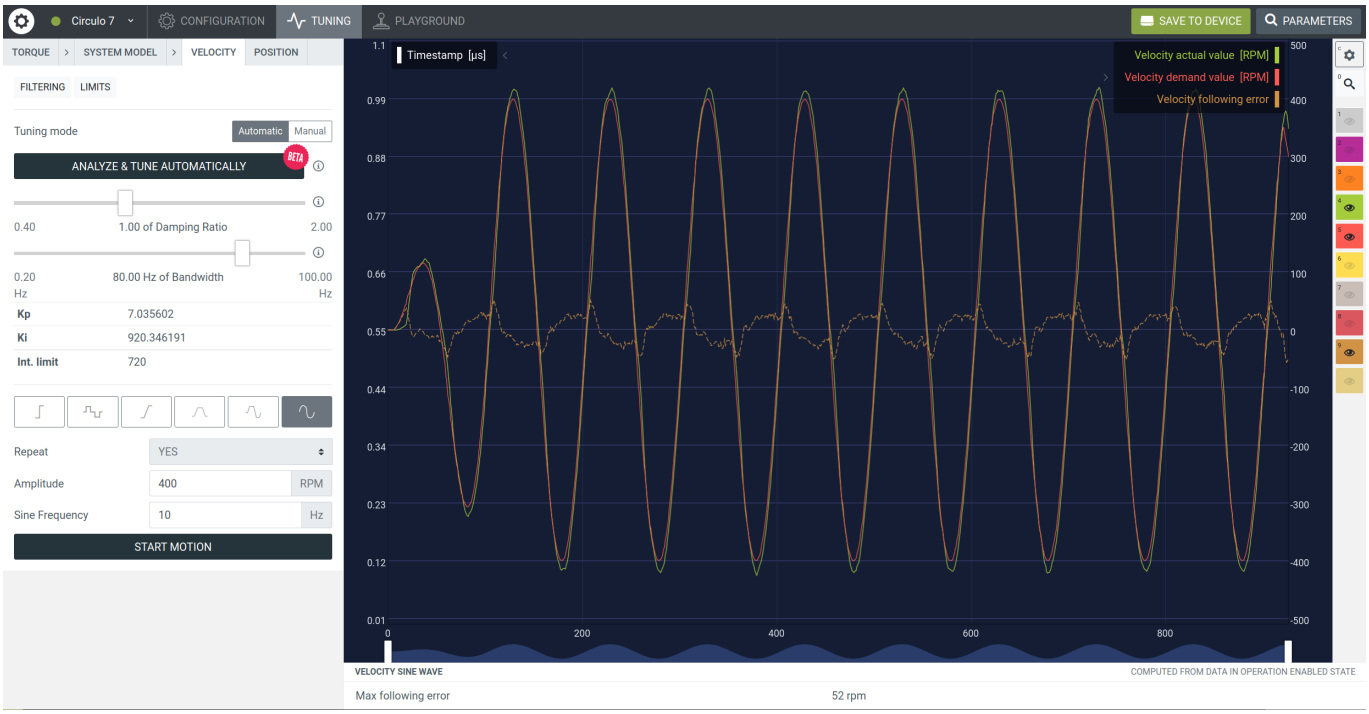
A damping ratio higher than 1 may lead to undershoot and non-zero steady state error.

- **Bandwidth** (also called **3 dB bandwidth**) is a characteristic of a control loop that characterizes sharpness of the control loop. It is a frequency the system may follow while

keeping 71% of the amplitude.Example of velocity control loop 10 Hz sine wave tracking with a requested bandwidth of 10 Hz:



An example of velocity control loop 10 Hz sine wave tracking with a requested bandwidth of 80 Hz:



Next steps

After tuning your motion control system, you can for example improve performance by using these features:

Cogging Torque Compensation

Removes ripples caused by magnetic interaction between motor rotor and stator.

Feedback signal filtering

Digital filters can be used to remove unwanted frequencies such as noise or nonlinear effects.

Gain scheduling

Applies controller gains which will change dynamically regarding the current speed.

F.A.Q.

Do I need to manually adjust controller gains after the auto-tuning?

- Usually, sliders provide enough flexibility to achieve demanded performance.

Can I define other velocity/torque units in the custom firmware?

- The auto-tuning will provide gains only for default units.

What if my encoder has low resolution?

- This will definitely result in worse performance compared to higher resolution encoders. Additionally, the velocity signal used in system identification might have multiple zeros that may result in incorrect model.

Do I have to repeat system identification and tuning after I attached a load to the system?

- It is better to repeat both procedures. If it is not possible to repeat the system identification, auto-tuning still can be used. But the resulting settling time will differ from the requested.

What if the system is unstable after auto-tuning?

- Try to input smaller bandwidth.
- Check the motor and sensor configuration parameters.
- Check the velocity signal in the playground in the torque control mode. Ensure that it shows correct data during manual rotation of the shaft.
- Soften filters in position/velocity control loops. They might introduce delay.

What if the test signals trigger overcurrent/undervoltage?

- Reduce the test signal amplitude or frequency.
- Reduce bandwidth.

- Check the configuration of the motor and sensor.
- Make sure that correct power supply is used.

How do filters influence the auto-tuning?

- Using a velocity filter results in a phase delay based on the filter order and its cutoff frequency. It is recommended to perform a system identification and tuning after filters were configured. Be careful, filters severely reduce stability due to phase delay.

Manual Tuning

Auto-tuning is the most convenient and in most cases sufficient way to get a sufficiently tuned overall system. However, all controller gains can also be adjusted manually.

- Manual Tuning of the current controller
- Manual Tuning of the velocity control loop
- Manual Tuning of the position control loop

Manual Tuning of the current controller

For a three-phase PM motor, the torque generated in the motor is directly proportional to the RMS value of motor phase currents.

The model-based feed forward controller is configured by motor parameters such as motor phase resistance and inductance.

There are two main challenges in implementing such controller in real-time systems:

- model mismatch
- sensitivity to the measured noise

To compensate model mismatch, an integrator is added to current loop controllers. The controllers are damped to reduce the sensitivity to noise.

Depending on the required bandwidth of the application, the user can change the default tuning configuration of the current loop controllers.

Manual tuning guide

The controller can be tuned similar to a PI controller. Controller constants (K_p and K_i) are of type integer, and their possible range is [0 - 1,000,000].

The default values of controller constants are K_p = 2,500 and K_i = 40,000. In most applications, these default values result in stable (but not sharp) controller behavior. In case the user wants to improve the tuning of current loop controllers, the following steps are suggested:

- Set K_i = 0 (and K_p = 2,500)
- Enable the controller in CST mode, and evaluate the torque (or current) controller step response. The current loop step response can be directly evaluated by using a current probe and an oscilloscope. In case OBLAC Drives is available, the torque controller step response can be evaluated by monitoring the “target torque” and “torque” in “tuning” tabs.

> Attention:
>
> In the entire procedure of tuning, the rotor should be blocked (fixed angle).

- Increase K_p by 50% and evaluate the step response. Repeat this step until the response reaches 80% of the reference value.
- Set K_i = 1,000.
- Increase K_i by 50% and evaluate the step response. Repeat this step as long as generated overshoot is below 10% of the reference value.

In case monitoring the step response (either by using a current probe and oscilloscope, or by using OBLAC Box) is not possible, the following rules of thumb can help in fine tuning the current loop controllers:

- Set K_p = 2,500 and K_i = 40,000
- Increase K_p by 20%, and send a step torque command with a period of 1 second and amplitude of 5% of rated value.
- Repeat this step until a vibration is felt in the system.
- This vibration means that the system is close to its instability margin. At this point reduce the K_p by 30%.

Manual Tuning of the velocity control loop

If the drive is intended to be operated in velocity control mode, the proportional and integral gains of the velocity control loop have to be tuned manually.

To do so, it is recommended to use OBLAC Drives to generate a velocity reference similar to a typical velocity reference of the intended application. This can either be a constant velocity or a dynamic signal (sinewave, trapezoidal etc.). While this reference trajectory is continuously being sent to the drive, you can increase the controller gains to optimize the tracking behavior. Proportional and Integral gains should both be iteratively increased. There are various procedures on

how to carry out this iterative process. As a general rule of thumb on which parameter to change next, the following is helpful:

- If the behavior of the system at low frequencies shall be improved, the Integral gain has to be changed. This applies if either it takes one or several seconds too long to reach a desired speed or if the speed oscillates between ‘too fast’ and ‘too slow’ slowly (once per second or even slower).
- If the behavior of the system at high frequencies shall be improved, the Proportional gain has to be changed. This applies if either acceleration and deceleration duration shall be reduced even further (to less than about 1 second) one if the speed shows undesired fast osculations which often become perceptible acoustically.

Further reference

Manual Tuning of the position control loop: Cascaded controller

The cascaded controller can be tuned manually, which can become necessary if:

- The auto-tuner cannot be run due to mechanical limitations, attached robot arms or other restrictions
- The auto-tuner fails or does not give sufficient results for a very specific application scenario
- For experimental purposes

The cascaded controller implements two PI controllers: one on the position loop and one on the velocity loop. The one on the velocity loop uses different parameters than the pure velocity controller (the velocity controller which is not part of the cascaded position controller). You can tune both independently.

To avoid instabilities caused by the cascaded controller, it is recommended to use only one of the integrators at a time: Either the integrator on the position loop or the integrator on the velocity loop. When the integrator on the position loop is used, the gain for the velocity integrator should be set to 0. This configuration is refered to as PI-P. When the integrator on the velocity loop is used, the gain for the position integrator should be set to 0. This configuration is refered to as P-PI.

These initial parameters can be used as a starting point for cascaded position control-loop:

- Position P: 0.05.
- Position I: 0 for P-PI configuration. 0.5 otherwise.
- Velocity P: 0.5;
- Velocity I: 0 for PI-P configuration. 0.5 otherwise.

Using these initial values, controller gains could be manually tuned using different methods. See: Cascaded PID controller tuning

The simple PID controller has generally poorer dynamic performance than the cascaded PID controller. However, it can be used for experimental investigations as the easiest configuration. Manual tuning is mandatory.

The following initial parameters could be used:

- Position P: 0.01
- Position I: 0.02
- Position D: 0.001

Using these initial values, controller gains could be manually tuned using your preferred methods. Classical tuning methods for PID controllers like Ziegler-Nichols can be applied here.

> Note:
>
> The PI-P cascade controller may look similar to the simple PID controller, however the structure is different and therefore a different tuning method should be used.

OBLAC Drives Setup

There are two ways to run SOMANET OBLAC Drives:

OBLAC Box: is a physical machine that comes preinstalled with Linux OS and runs OBLAC Box.

OBLAC Drives Virtual Machine: comes preinstalled with OBLAC Box. It can be run on Windows, Linux and macOS.

If you run into any problems, this might be ale to help you:

Troubleshooting: If something goes wrong while using OBLAC Drives you can find troubleshooting instructions here

OBLAC Box

> Important:
>
> In case you are not using OBLAC Box, please refer to our OBLAC Drives virtual machine installation guide.

OBLAC Box



OBLAC Box 2

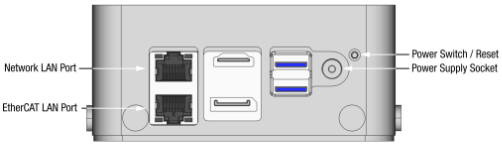


OBLAC Box is a physical machine that comes preinstalled with Linux OS and runs OBLAC Drives and the related services. It supports access over WiFi or local area network.

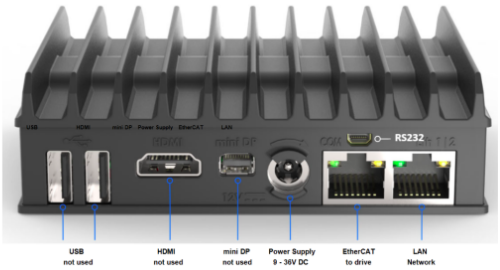
WiFi Frequency	2.4/5 GHz (Antenna, Dual Band 2.4/5 GHz, Rubber Duck, RP-SMA)

Connectors

OBLAC Box



OBLAC Box 2



Installation

OBLAC Box

> Note:
>
> The power supply has an output rating of 5 V_{DC}, 4000 mA.

OBLAC Box 2

> Note:
>
> The power supply has an output rating of 12 V_{DC}, 3000 mA. OBLAC Box 2 can be supplied with 9-36 V_{DC}, ≥15 W

1. Plug in the power connector; the box will automatically boot.
2. Connect one or multiple drives to the EtherCAT port 1 on the box
3. Establish a network connection to the box by Wi-Fi or per Network Cable (see Network Connection to the OBLAC Box)
4. Open a browser (Google Chrome preferred) and connect to the box (see Open OBLAC Drives)
5. The OBLAC drives dashboard will appear
6. Update the box with the latest available firmware (see Update OBLAC Drives)
7. All connected SOMANET drives should be visible. Select the Servo Drive you want to configure and tune

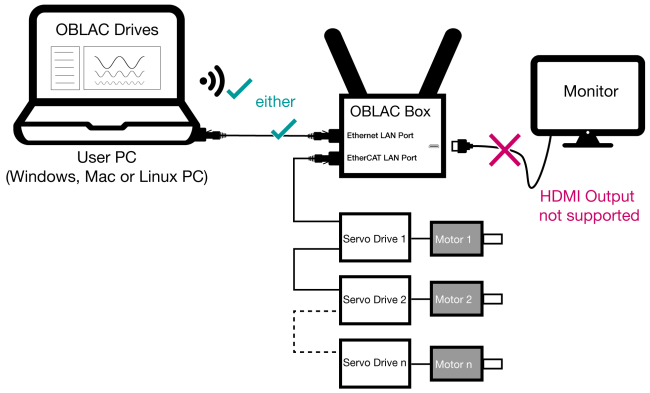
Attention:

Ensure that the LAN ports are used correctly: The servo drives have to be connected to the EtherCAT LAN port. If the ports are mixed up, no communication to the drives and no connection to the box via LAN is possible.

Note:

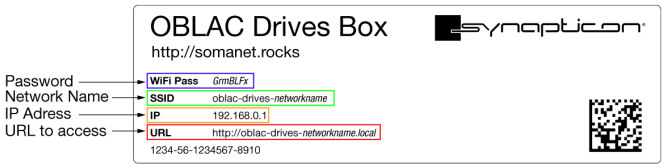
The HDMI port is without function. You require a dedicated host PC to control OBLAC Box.

Multiple instances of a browser connection to a box can be established. A box can operate multiple connected drives independently, but one browser tab can only operate one drive. You can even connect to one drive on multiple browser instances (e.g. operate and get diagnostic information at the same time). But the last action will override the previous. There is no interlock between the instances.

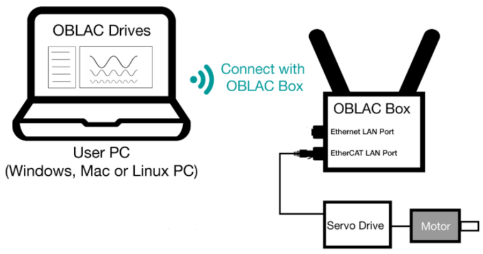


Network Connection to the OBLAC box

Depending on different needs, several ways to establish a network connection are possible. In each case the drives have to be connected to the EtherCAT port of the box. Some important device specific information is written on the sticker placed on the box:

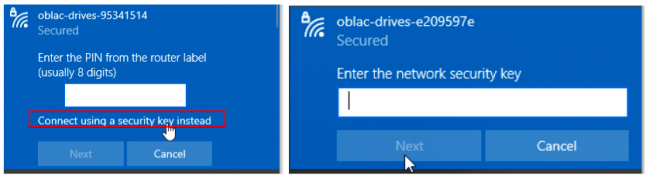


Connection Host-PC ↔ Box by Wi-Fi



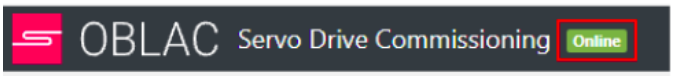
> Note:
>
> This is the preferred way to connect to an OBLAC Box.

To establish the connection you have to activate the Wi-Fi on your host PC and connect to the Wi-Fi with the network name written on the box. Each box has a unique SSID. Open the Wi-Fi configuration of your PC, select the given network and enter the password. On Windows 10, you have to enter the password as security key:



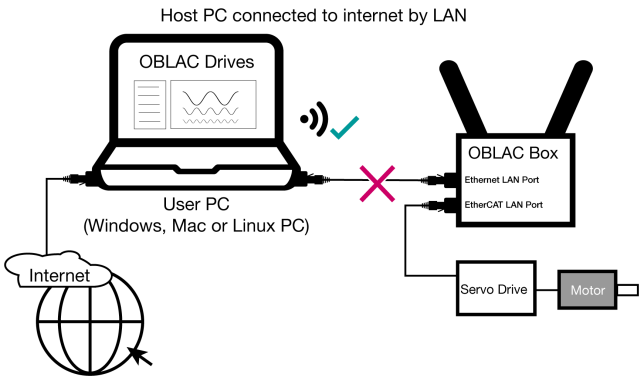
After the Wi-Fi connection is established, you have to open a browser (Google Chrome preferred) and enter the IP address given by the label in your browser's address bar (always 192.168.0.1). Alternatively you can enter the URL given on the label of the box. The OBLAC Drives dashboard will appear. You can now operate connected drives with OBLAC Drives.

To be able to do any updates, a connection to the internet or a previously saved bundle is required. Different ways to realize this connection are described in the following chapters. A connection between the box and the internet will be reported by the green "online" sign in the top line of the OBLAC Drives page.



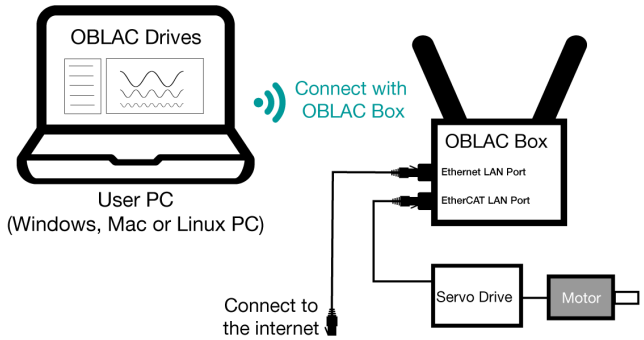
Host-PC connected to Internet by LAN connection

If the host PC is connected to the internet by a LAN connection or a second Wi-Fi connection, the box will automatically connect to the Internet through the host PC.



Box connected to internet by LAN connection

If there is no connection to the internet with the host PC possible, you can alternatively connect the OBLAC box to a network by LAN network port on the box. On this port, DHCP is configured and a DHCP server is required to give the box an IP-address.



> Attention:
>
> In some companies, unknown devices will not get an IP address from the server out of security reasons. In this case this will not work. Ask your IT/network responsible if you fail to get online with the box.

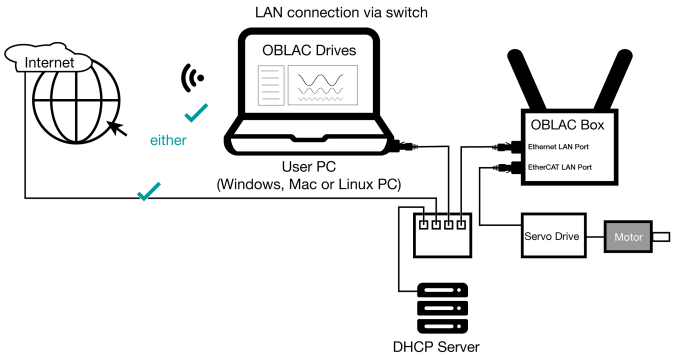
Host-PC and Box not connected to Internet

If neither the PC nor the box can't be connected to the internet, the offline update by bundles can be used (see Offline Updates).

Connection Host-PC ↔PC by LAN cable

If the Wi-Fi connection of the host PC is already used for accessing the company network / internet connection, the connection to OBLAC Drives can be done by connecting the host PC to the OBLAC box by a LAN cable. Use the LAN / Network port in this case.

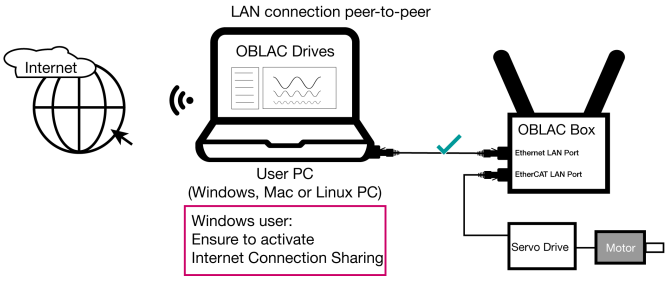
LAN connection via Switch and connection to a network with DHCP server



If the box and the PC are connected to a LAN that includes a DHCP server (regularly by connecting to a switch), the box and the PC port will get an IP address from the DHCP server in the same range. To connect to OBLAC Drives, it is not needed to know this IP-Address. It is enough to call the local URL of the Box printed on the label of the box (http://oblac-drives-xxxxxxxx.local)

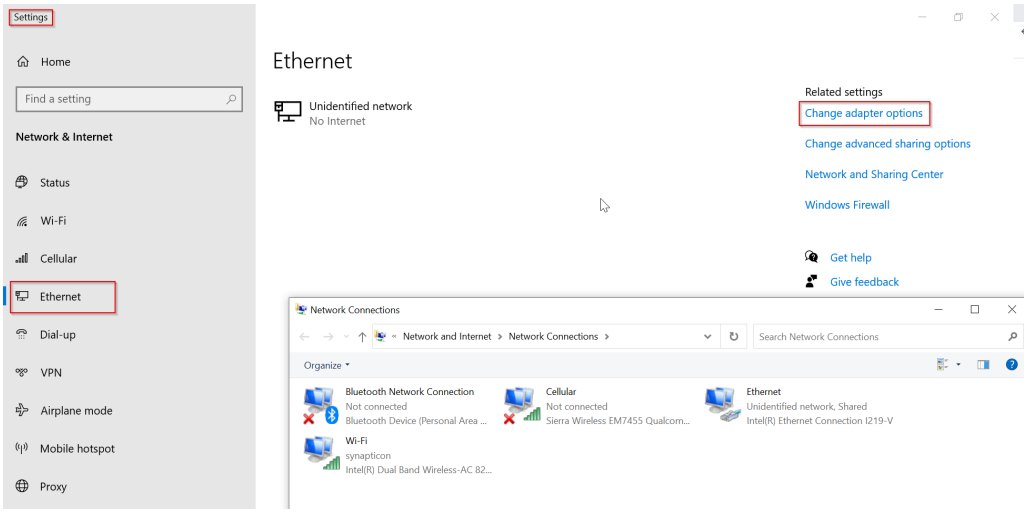
> Attention:
>
> In some companies, unknown devices will not get an IP address from the server out of security reasons. In this case this will not work.

LAN connection Peer to Peer

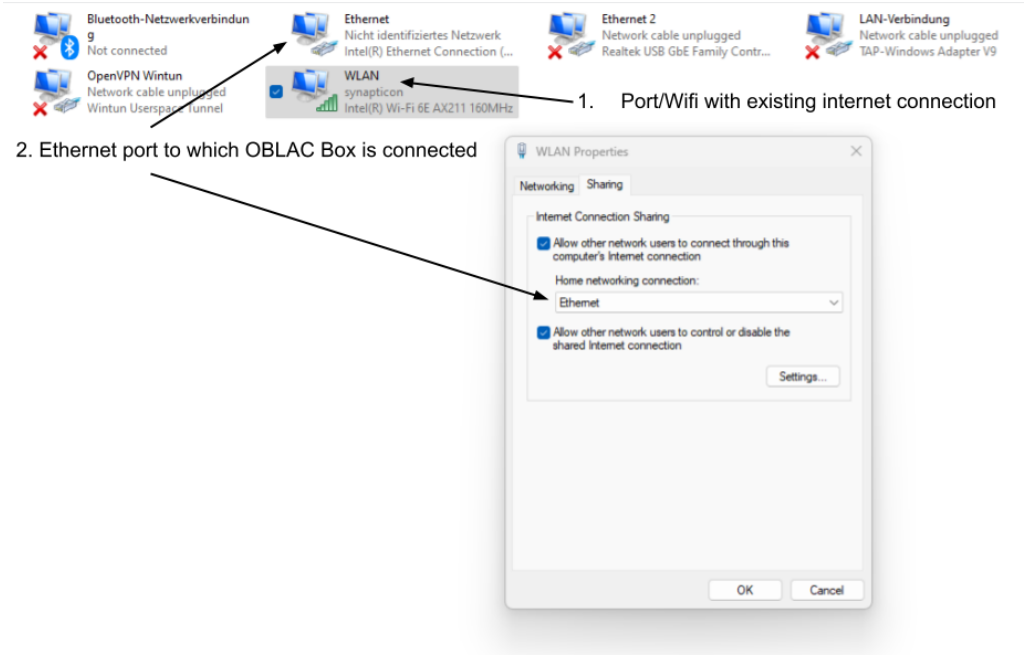


If there is no DHCP server available, a single peer to peer connection can be used. Just plug in an Ethernet patch cable between your PC board and the LAN port of the OBLAC box. But in addition ICS (Internet Connection Sharing) has to be enabled on the PC.

Go to Windows Settings - Network & Internet:



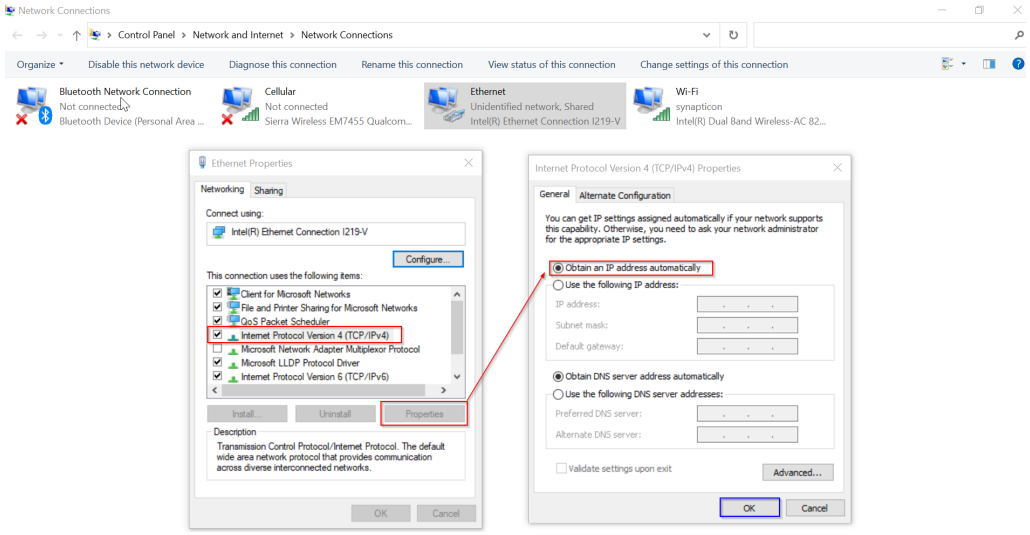
- Right-click on the Ethernet Port you are connected to the internet.
- Open properties and Sharing tab.
- Enable the first check box.
- Select the Ethernet port your OBLAC box is connected to.



Now you can connect to the box and open OBLAC Drives out of the browser by connecting to http://oblac-drives-xxxxxxxx.local. You find the box specific address on the label on your box.

The first time after activating the ICS, it might take up to one minute till the bridging is done and you can connect to OBLAC Drives.

The ICS is not working if the IP-address is manually set to a fixed address in the PC. Please ensure that the port is configured to “obtain IP address automatically”.

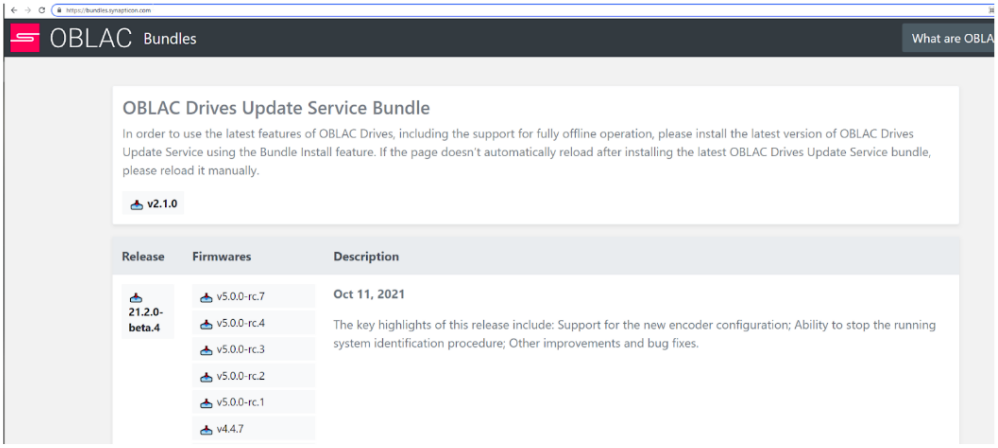


No connection to OBLAC update server

The box retrieves updates from an update server in the cloud. Some company firewalls block those connections. Ask your IT responsible if you get fault messages in not being able to connect to the update server. You can either remove those blockings or continue with offline updates (see below).

Offline Updates

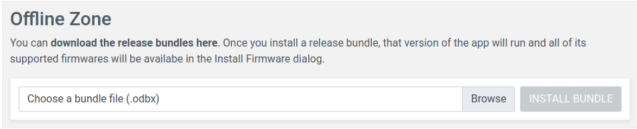
If it is for any reason not possible to get online with the OBLAC box, the offline update with “bundles” is possible. The bundles can be downloaded from https://bundles.synapticon.com/.



There are two types of OBLAC Bundles:

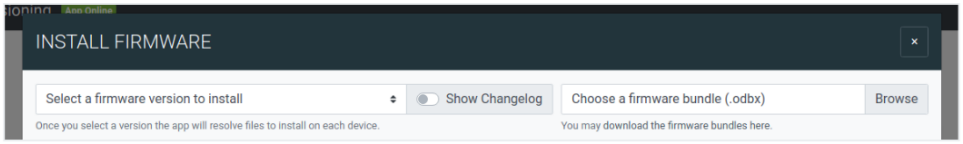
1. Release bundle

   This bundle includes an OBLAC Drives release, e.g. v21.0.0 and all of its supported firmwares. Install it using the Bundle Install feature in OBLAC Drives Update Service. Once installed, the selected release will run, and all of its supported firmwares will be available in the OBLAC Drives Install Firmware dialog.



2. Firmware bundle

   This bundle includes files for all hardware variants of a single firmware version, e.g. v4.4.3. Install it using the OBLAC Drives Install Firmware dialog onto one or more devices.



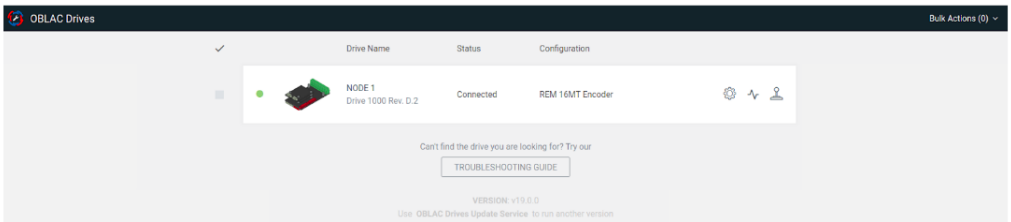
> Note:
>
> Offline update is available from OBLAC Drives version 21.0.1. upwards

Open OBLAC Drives

Open OBLAC Drives in your browser by navigating to http://oblac-drives-networkname.local

> Note:
>
> Windows users need to install the Bonjour Software to open OBLAC Box at the given URL.

If this is not possible you can still access OBLAC Drives via Wi-Fi by entering the IP address given on the sticker. Please check in your browser that the Drive is connected:



Update OBLAC Box

Before using OBLAC Box, you need to install the latest version of OBLAC Drives with our OBLAC Drives Update Service.

It is possible to use the OBLAC Box offline and to install and run the previous versions of OBLAC Drives. Newer versions of OBLAC Drives that get installed will be cached on the box, but they will have to be downloaded from the Internet.

OBLAC Drives Virtual Machine

> Important:
>
> This section is only required if you are **not** using OBLAC Box.

Prerequisite

- OBLAC Drives Virtual Machine is run on a virtualization software. VMware Player version 14 or above for Windows and Linux or VMware Fusion 10 for macOS are needed for running OBLAC Drives Virtual Machine.

> Note:
>
> OBLAC Drives is an application that runs in a Web Browser. We recommend the latest version of Google Chrome.

Getting started

1. Download the OBLAC Drives Virtual Machine image

DOWNLOAD OBLAC DRIVES

If you have trouble downloading from Amazon webservice, please try our Baidu server:

DOWNLOAD OBLAC DRIVES FROM BAIDU

Windows

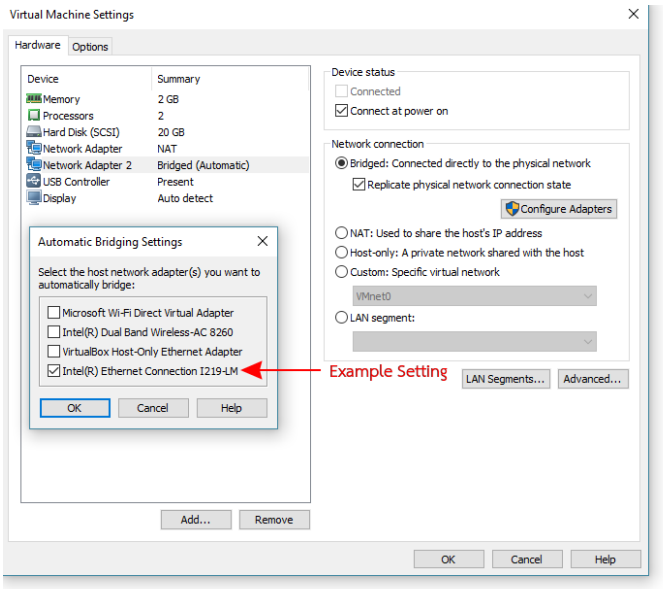
2. Install and open VMware Workstation Player

> Note:
>
> ANY version of VMWare Workstation Player starting from 14.0 can be used for OBLAC Drives.

3. Open a virtual machine
4. Select the downloaded OBLAC Drives image (OVA file) and import it to VMware Player

> Note:
>
> If you get a warning that the import failed, just click “Retry” and it will run.
>
> You may also check “Do not show this message again”

5. Go to Virtual Machine Settings and check if two network adapters are available - one set to NAT and one set to Bridged (Automatic).
   - Ensure that the "Replicate physical network connection state" option is checked.
   - Press OK to save changes.



6. Power up and connect the slaves.
7. Power on the virtual machine

macOS

2. Install and open VMware Fusion

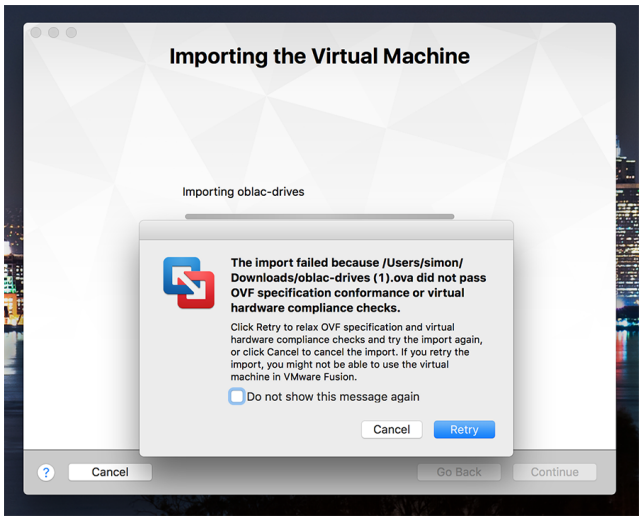
> Note:
>
> ANY version of VMWare Fusion starting from 10.0 can be used for OBLAC Drives.

3. Open the OBLAC Drives image (OVA file) in VMware Fusion

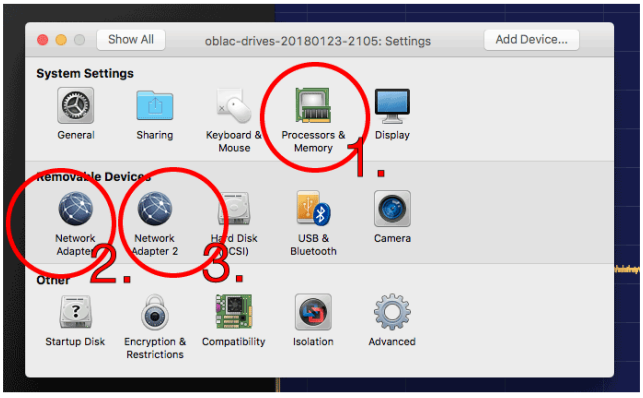
Note:

If you get a warning that the import failed, just click “Retry” and it will run.

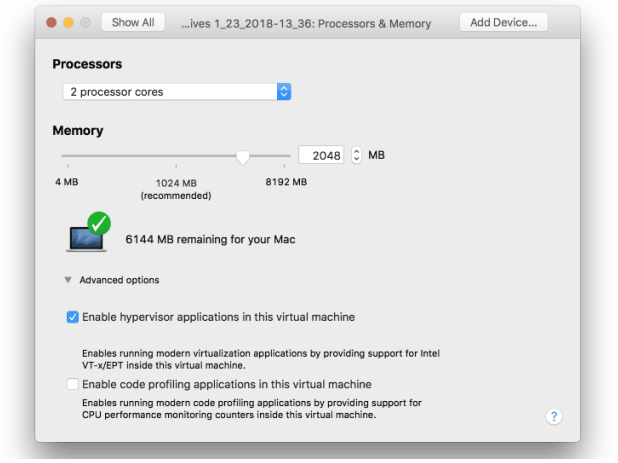
You may also check “Do not show this message again”



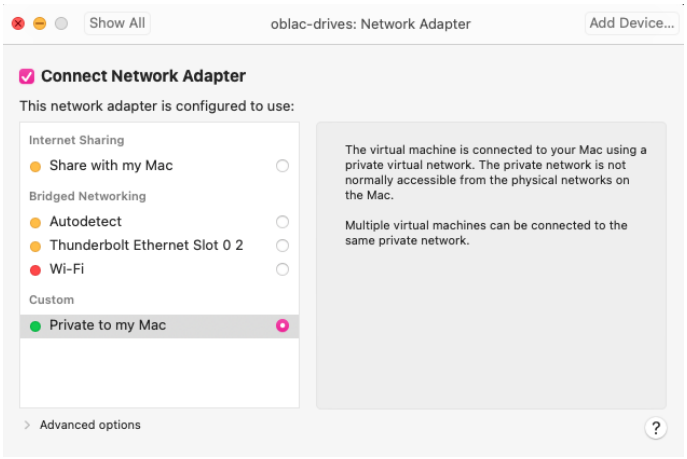
4. Edit the Virtual Machine Settings and set the following (if not already set)



- Set the number of **processor cores** to 2
- Check the “Enable hypervisor application” option



- The **network adapters** should be set to: *Private to my Mac*



5. Power up and connect the slaves.
6. Play the virtual machine.

Linux

2. Install and open VMware Workstation Player

> Note:
>
> ANY version of VMWare Workstation Player starting from 14.0 can be used for OBLAC Drives.

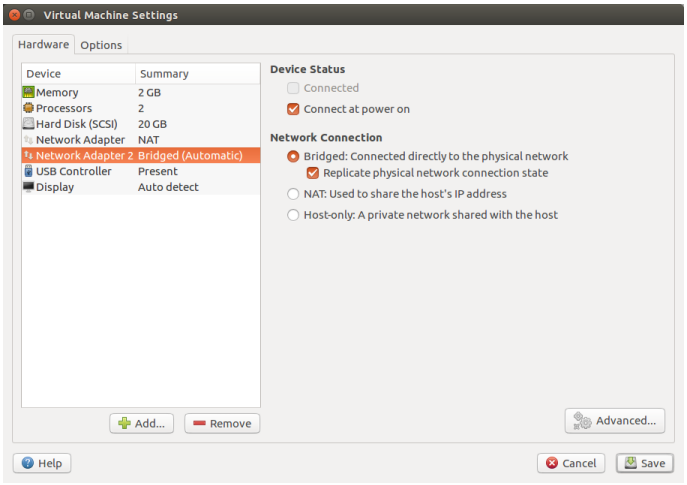
3. Open a virtual machine:

   Select the downloaded OBLAC Drives image (OVA file) and import it to VMware Player

Note:

If you get a warning that the import failed, just click “Retry” and it will run. You may also check “Do not show this message again”

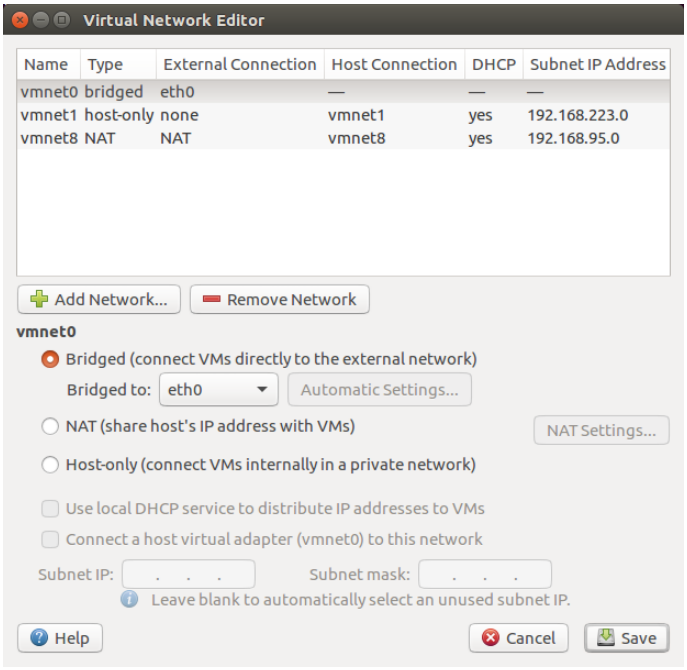
4. Go to Virtual Machine Settings and check if two network adapters are available - one set to NAT and one set to Bridged (Automatic)
   - Ensure that the “Replicate physical network connection state” option is checked
   - Press OK to save changes



5. Network settings
   - Execute

```
sudo /usr/lib/vmware/bin/vmware-netcfg
```

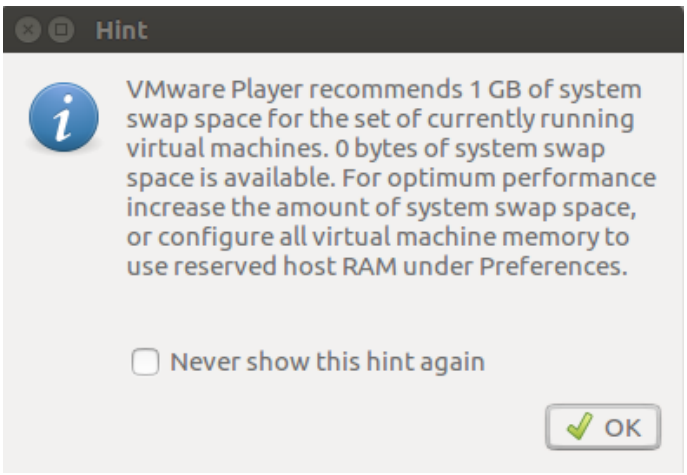
   - Select the bridged network adapter and bridge it to the system network adapter to which the slaves are connected to
     - The network adapter has to be an Ethernet connection, so it likely starts with “eth”
   - Save changes



6. Power up and connect the slaves.

To connect your host PC with the servo drive, simply plug the EtherCAT cable on your Ethernet port (RJ45) of your PC and connect it to the servo drive on EtherCAT port IN.

7. Play the virtual machine
8. In case your systems hints that a swap partition is needed:



instead of creating such a partition you can also use this workaround:

- Close the VMware Player
- Run

```
sudo gedit /etc/vmware/config
```

- Add this line at the end of the file and save:

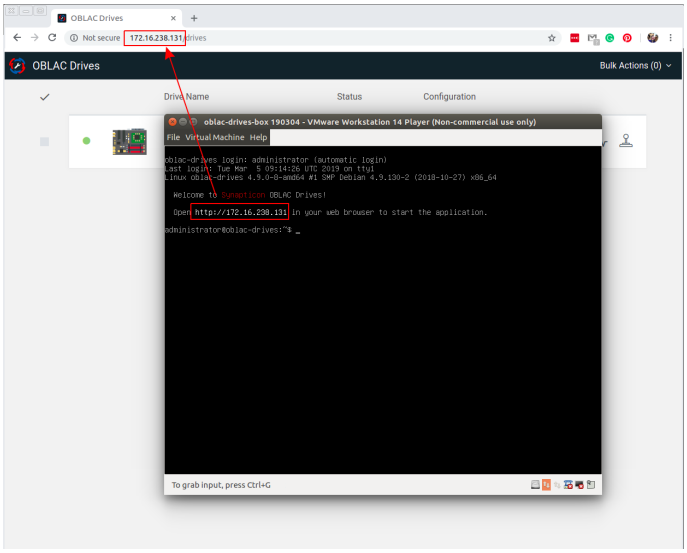
```
prefvmx.minVmMemPct = "100"
```

- at the end of the file and save

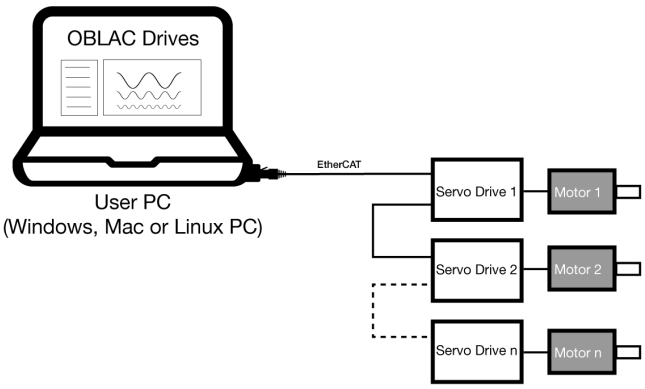
  Restart the VMware Player

Connecting to OBLAC Drives

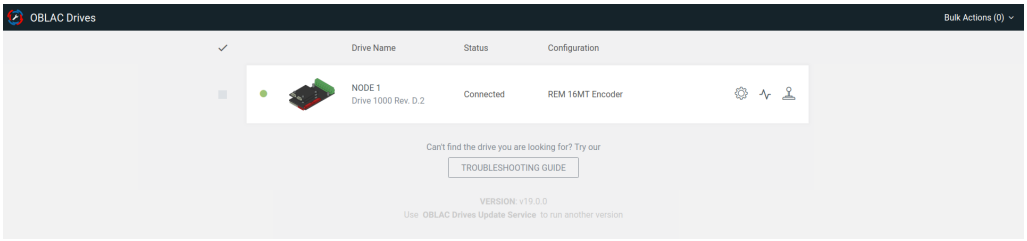
Once your virtual machine is powered up, you will find the IP address of OBLAC Drives at the top of the screen. Enter this IP address in your browser and wait for OBLAC Drives to show the connected drives.



Connect your PC via EtherCAT cable to the servo drive. For tuning, you may connect several servo drives via EtherCAT and tune one by one.



Check in your browser that the Drive is connected:



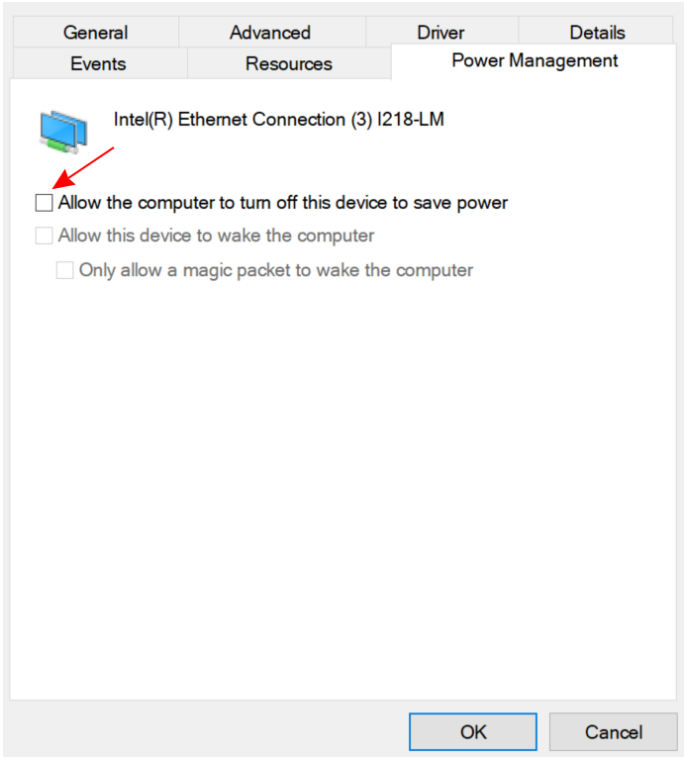
Update OBLAC Drives

Before using OBLAC Drives, you need to install the latest version of OBLAC Drives with our OBLAC Drives Update Service.

It is possible to use OBLAC Drives offline and to install and run the previous versions of OBLAC Drives. Newer versions of OBLAC Drives that get installed will be cached on the box, but they will have to be downloaded from the Internet.

Troubleshooting

1. Always give OBLAC Drives up to a minute to detect the available drives after they have been powered on or connected
2. If the drives aren't found, or anything else seems to be wrong, please try refreshing the web page first and then power cycling the drive(s)
3. If you are running Windows, please ensure that the power-saving functionality for LAN is turned OFF



4. In some cases after the start-up of OBLAC Drives the URL isn't shown and the website can't get accessed. In that case, please try switching the order of the network adapters in the VMware setting for that virtual machine:
    - Bridged (Automatic)
    - NAT

There are more issues listed in our Troubleshooting Guide

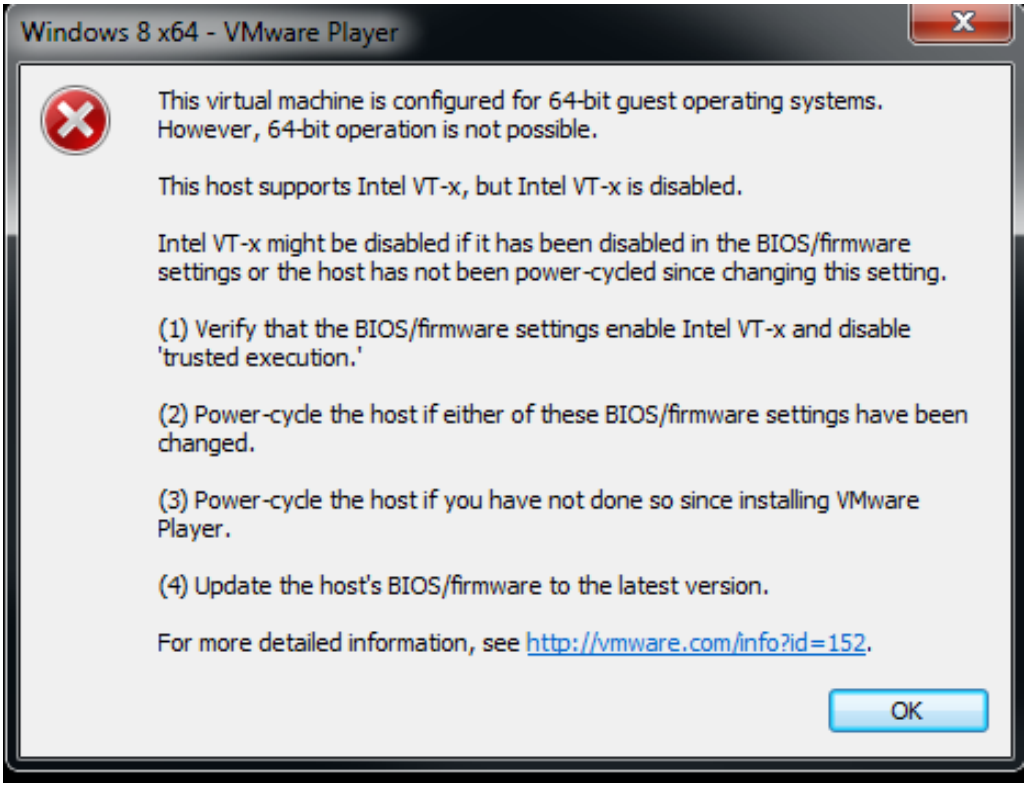
Troubleshooting

This section shall guide you through the most common error scenarios that could occur when using OBLAC Box.

> Important:
>
> Make always sure to carefully read through the setup instructions. A lot of issues can be avoided by strictly following the instructions.

Virtualization not activated

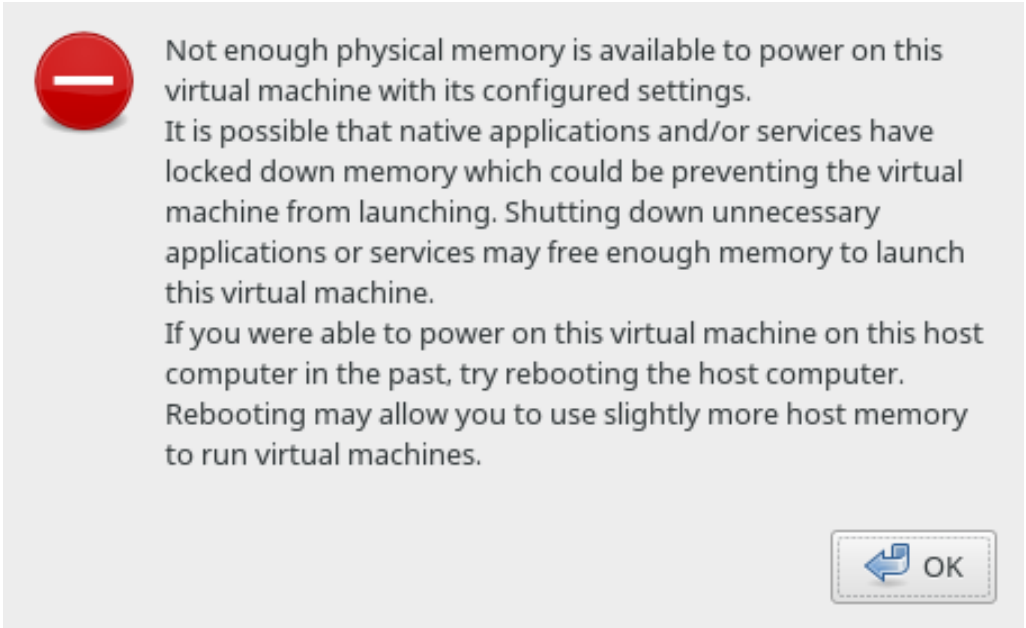
If your computer does not have the virtualization technology activated, OBLAC Drives will most likely not work correctly.



To fix that issue, you have to activate the virtualization technology of your processor (VT-x / AMD-V) in your BIOS.

Memory Warning

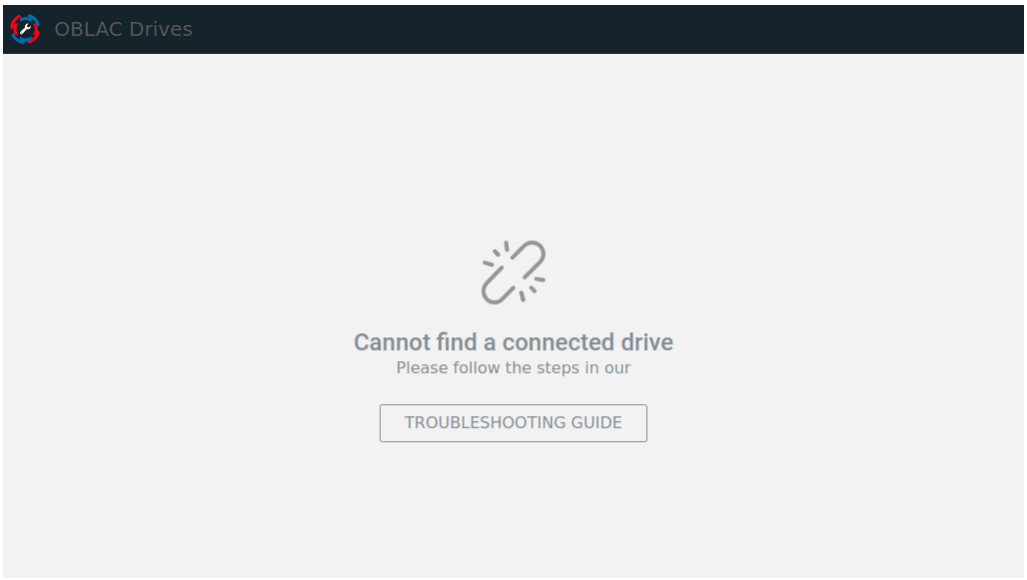
In rare cases your Linux distribution may show the warning: Not enough physical memory available to power on this virtual machine”



To fix this issue, follow the commands given here

Runtime Issues

One of the most common issues appears directly after installing OLAC Drives. The browser will be opened and the following screen appears:



There are several reasons that will inevitably lead to this problem:

- you did not power on the drive before starting OBLAC Drives
- you selected the wrong network port while configuring the VMware settings (check out our Setup Guide)

Other Issues

If the application does not behave properly in any other way than described above, try the following:

- Refresh the OBLAC Drives web page
- Make sure no native EtherCAT master is running on the host machine or in the same network
- Make sure that none of the Synapticon tool components (OBLAC Drives, motion master bridge or motion-master) are already running on the host machine as a native installation
- If your laptop is connected to a docking station, undock and restart it, then try to run OBLAC Drives again
- Other installed or running software on the PC is occupying the Ethernet port.

  If possible, stop or uninstall this software or try it with a fresh PC.

- Not all network adapters are able to perform the real-time requirements for OBLAC Drives. In our experience, network cards inside laptops and computers work most reliably. Docking stations (if not connected via USB) are also often reliable.

  A network connection via USB is difficult to predict. Some adapters work very well, others not at all. If USB is the only option, you may have to try several adapters or even a different PC. This is most likely the cause if you can establish a connection to a drive, but the connection is interrupted frequently.

- Limited performance of the host PC. If the general performance of the PC is too weak (less than 8 GB RAM, low frequency) or too many other software is running on the PC, there is not enough power left for the virtual machine.

  Stop all other software that is running or try with a different, more powerful PC

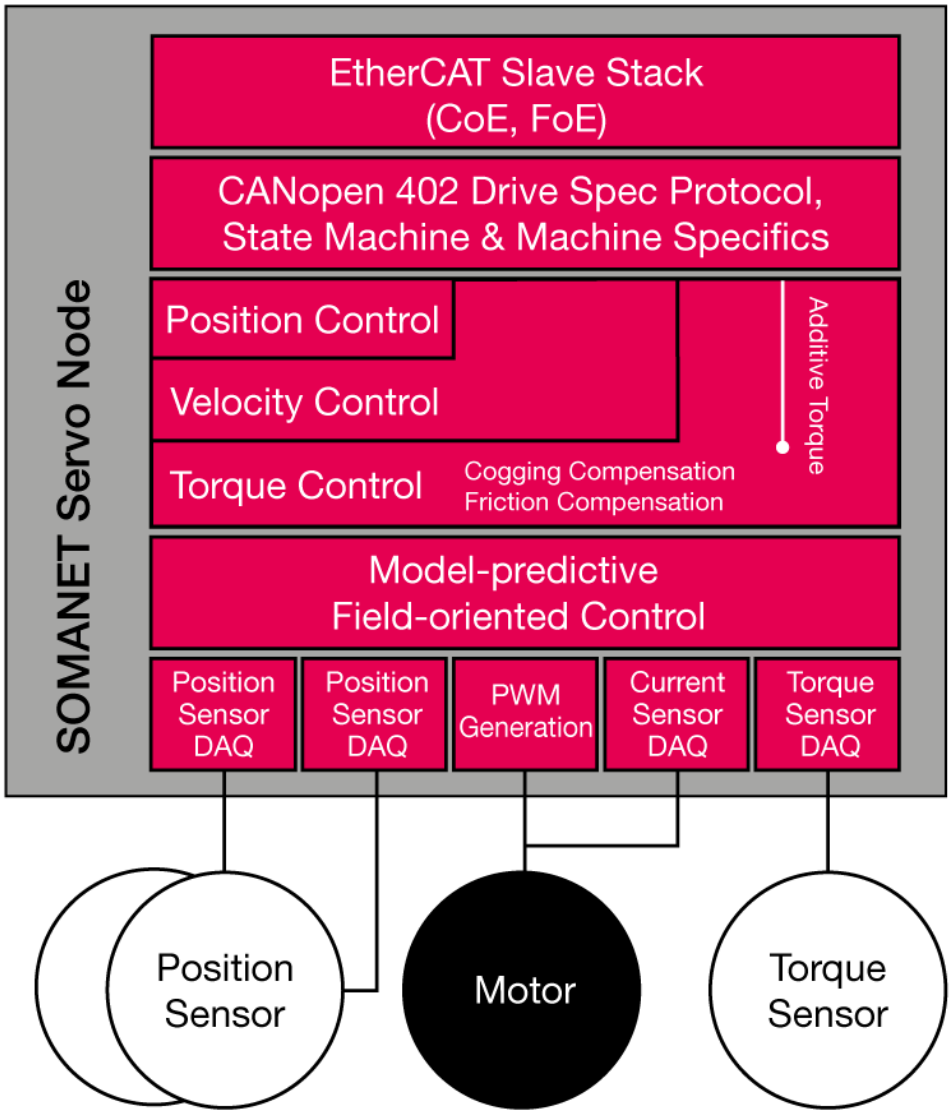
> Note:
>
> If you couldn't find a solution to your problem, please contact us at support@synapticon.com.

Firmware Overview

All SOMANET servo drives support both cyclic synchronous modes of operation **CSP, CSV, CST** optimized for use in robotic applications. and profile modes **PP, PV, PT** for point-to-point applications. Firmware updates and storing of configuration files is possible via File over EtherCAT **(FoE)**. Multiple concurrent encoders (for example **BiSS-C, SSI, Incremental, Digital Halls**) are supported.

An object dictionary according to **CiA 402** is available, complemented with manufacturer specific objects to expand the feature set.

The following diagram depicts the basic architecture of the servo drive system:



Functionality description

Cyclic Synchronous Modes:

Cyclic synchronous torque control (CST) In torque control mode, the master sends desired torques to the drive

Cyclic synchronous velocity control (CSV) In velocity control modes, the master sends desired velocities to the drive

Cyclic synchronous position control (CSP) In position control modes, the master sends desired positions to the drive

Profile Modes:

Profile position mode In profile position mode, the trajectory to a target point is generated on the drive

Profile velocity mode In profile velocity mode, the motor will rotate at a target velocity

Profile torque mode In profile torque mode, the drive sends a target torque to the motor

Homing Modes that support limit and home switches

Set of control architectures:

- Field-oriented current control
- PI and model-predictive current control
- P, PI, PID and cascaded position control
- P, PI, PID velocity control
- Dual loop cascaded position control
- Controller Auto-Tuning and manual tuning

Various configurable signal processing options (low pass filters, top notch filters)

Encoders configuration for all ports: BiSS-C, Hall, ABI (QEI) / ABZ Encoder, SSI , A-Format

Access to Analog Inputs and Digital IO

Digital I/Os can be configured to timestamp inputs and outputs on rising edge

Optimized power output by 3rd harmonic injection

Synchronization with Distributed Clocks

Simple commissioning and tuning with OBLAC Drives

Extended control functionalities:

Cogging compensation Removes ripples caused by magnetic interaction between motor rotor and stator

Field weakening The velocity range of electric motors can be extended by weakening the magnetic field of the rotor linearly over the speed range

Anti-windup control An Anti-Windup algorithm is used to prevent the Integrator part to accumulate beyond acceptable boundaries

Gain scheduling Applies controller gains, which will change dynamically regarding the current speed

Motor overload protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating

Velocity feed forward To increase the velocity of the robot's reaction, the target position signal can bypass the position controller

Torque offset A desired offset torque can be added to the reference torque

Velocity offset An additional velocity offset can be used to improve the positioning performance

Filtering Digital filters can be used to remove unwanted frequencies such as noise or nonlinear effects

Position and Velocity Feedback Low-pass filters

Torque loop input shaping filter

Command smoothing and Interpolation Successive setpoints can be smoothed, so that low command resolution and jitter have a reduced effect on the control

Control supervision Allows to supervise the work of the control loops

CANopen over EtherCAT (CoE)

CANopen is a fieldbus system and device profile specification for embedded systems used in automation. Among others CANopen defines communication models like master/slave, where the master sends or requests data from the slaves. CANopen over EtherCAT (CoE) incorporates specialized profiles for motion control and supports a standardized set of commands to be used for motion control. This set of commands is known as CiA 402 and is defined in the IEC standard IEC 61800-7-201:2015 for adjustable speed electrical power drive systems.

If you are using TwinCAT as your controller, you will need to import the SOMANET CiA 402 ESI (EtherCAT Slave Information) XML file to your project.

Drive State Machine (CiA 402)

Each CiA 402 device is controlled by the State Machine.

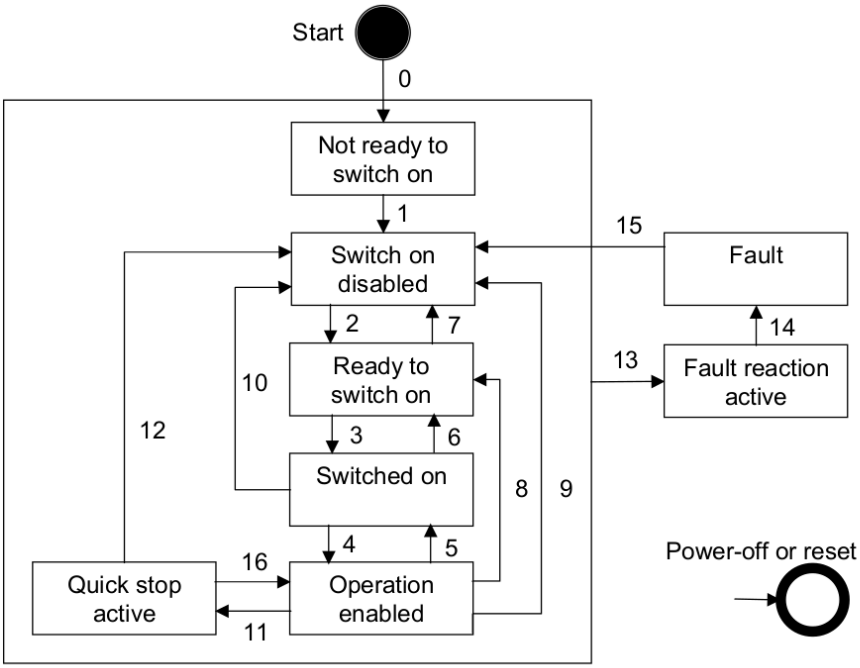


Image 1 - The transitions of the system's states according to IEC 61800-7-201

The function that are available in each state are given in the table below:

	FSA States							
Function	Not ready to switch on	Switch on disabled	Ready to switch on	Switched on	Operation enabled	Quick Stop active	Fault reaction active	Fault
Brake applied, if present	Yes	Yes	Yes	Yes	No	No	No	Yes
Drive function enabled	No	No	No	No	Yes	Yes	Yes	No
Con-	Yes	Yes	Yes	Yes	No	No	No	Ye-

	FSA States							
Function	Not ready to switch on	Switch on disabled	Ready to switch on	Switched on	Operation enabled	Quick Stop active	Fault reaction active	Fault
figuration allowed								s

The devices of the motion control system are configured by the Service Data Objects (SDO) while the real-time process data is transferred by Process Data Objects (PDO).

PDO

PDO (Process data objects) are used in EtherCAT to cyclically transfer in real-time control (e.g. target velocity) and status (e.g. current velocity) variables. These variables are identified by their Index and Subindex in the object dictionary. With the help of the PDO mapping, the objects from the object dictionary are assigned to either receive PDO (RxPDO, Master to Slave direction) or transmit PDO (TxPDO, Slave to Master direction). The standard mapping can be changed to a custom mapping to accommodate specific needs.

PDO data is internally updated by the slave (servo drive) at a rate of up to 4 kHz. The master can access the drive at 4 kHz maximum.

For more information about PDO refer to ETG1000.5 and ETG1000.6

PDO Configuration

SOMANET supports EtherCAT PDO Configuration for up to 20 objects on 4 maps. This allows to use up to 80 objects per direction.
PDO Config allows to change the mapping of the PDO mappings itself. Every PDO mappable object from the object dictionary can be added into a PDO mapping. It is possible to add up to 20 objects to one PDO mapping

SDO

SDO (Service Data Objects) are used to exchange parameters between master and slave devices in a EtherCAT Network. This isn't done in real-time. They provide read and write access to all entries in the device object dictionary.

Typical examples are:

- Configuring the Drive system
- Reading error codes

For more information about SDO refer to ETG1000.5 and ETG1000.6.

Actuator configuration

Motor and gear settings Description of the motor and gear parameters

Generic brake output voltage Phase D can be used for a single phase device or custom brake options

Brake configuration Description of the brake release feature

User defined velocity units Users can change the default velocity unit for higher precision

Commutation Offset Detection Aligns the motor's electrical angle with the position sensor feedback for motion control

Power limit Limits the mechanical output power of the motor

Motor and gear settings

Motor Configuration

The servo drive needs several motor parameters to operate correctly. They can usually be found in the motor data sheet and are entered in the corresponding objects via the "OBLAC Drives" configuration wizard.

Pole pairs

In a three-phase PM motor, the number of poles depends on the rotor design and shows how many "south" and "north" magnetic poles exist. These magnetic poles are created by permanent magnets which are built into the rotor structure.

This number can be found in the motor's data sheet.

Attention:

If the number given in the data sheet is not designated as Pole pairs but Poles, it must be converted:

$$\text{Number of pole pairs} = \frac{1}{2} \cdot \text{Number of poles}$$

Torque constant

The torque constant is the ratio between generated torque and the phase current amplitudes. The torque constant unit is in [µNm / A_{rms}]. The motor torque constant can be found in the motor's data sheet.

If in the motor data sheet, the unit of torque constant is in [Nm / A_{peak}], the following equation can be used to convert the unit of torque constant to [µNm / A_{rms}]:

$$K_t[\frac{Nm}{A_{rms}}] = \sqrt{2} \cdot K_t[\frac{Nm}{A_{peak}}]$$

Phase resistance

For a three-phase motor with star winding configuration, the phase resistance represents the resistance between motor terminals and the motor neutral point. If only phase to phase resistance is available in the motor data sheet, the value for phase resistance is half of the value phase to phase resistance in the motor data sheet:

Attention:

$$\text{Phase resistance} = \frac{1}{2} \cdot \text{Phase to phase resistance}$$

Phase resistance of delta winding motors

The software assumes that the motor windings are in a star configuration. For a "delta" winding configuration, the values which should be entered in the configuration must be converted to the equivalent star winding configuration.

If the value for phase resistance is available in the motor data sheet, the motor phase resistance for the equivalent star configuration equals to:

$$\text{Phase resistance}_{\text{equivalent star configuration}} = \frac{2}{3} \cdot \text{Phase resistance}_{\text{motor data sheet}}$$

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

Phase to phase resistance of delta winding motors

If the value for phase to phase resistance (also designated as terminal to terminal resistance or line to line resistance) is available in the motor data sheet, the motor phase resistance for the equivalent star configuration equals to:

$$\text{Phase resistance}_{\text{equivalent star configuration}} = \frac{1}{2} \cdot \text{Phase to phase resistance}_{\text{motor data sheet}}$$

Phase inductance

For a three-phase motor with a star winding configuration, the phase inductance represents the inductance between the motor terminals and the motor's neutral point. If only phase to phase inductance is available in the motor data sheet, the value for phase inductance is half of the phase to phase inductance value in the motor data sheet.

> Attention:
> $$\text{Phase inductance} = \frac{1}{2} \cdot \text{Phase to phase inductance}$$

Phase to phase inductance of delta winding motors

If the value for phase to phase inductance (also designated as terminal to terminal inductance or line to line inductance) is available in the motor data sheet, the motor phase inductance for the equivalent star configuration equals to:

$$\text{Phase inductance}_{\text{equivalent star configuration}} = \frac{1}{2} \cdot \text{Phase to phase inductance}_{\text{motor data sheet}}$$

> Note:
> Sometimes motor data sheets do not provide the phase inductance values but instead they provide the inductance values in direct and quadrature directions (Ld and Lq). In this case, the value of phase inductance can be calculated from the following equation:
> $$\text{Phase inductance} = \frac{1}{3} \cdot (\text{Ld} + \text{Lq})$$

Errors in phase resistance and inductance

Wrong values of the parameters phase resistance or phase inductance will affect the quality of the current loop controllers:

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

- narrower bandwidth
- less stability margins

> Note:
>
> Internal algorithms prevent degradation of the steady state operation when the error is less than 20% of the exact value.

Motor phases inverted

This parameter depends on the order in which the motor terminals are connected to the inverter terminals. It can be either “normal” or “inverted”. Details can be found in Commutation Offset Detection. This parameter is detected automatically when the sensor offset measurement procedure is triggered.

If Motor phases inverted is set wrong (manually), the following problems might appear:

- Wrong motion polarity (positive “torque target” resulting in negative “motor generated torque”).
- Low efficiency of motor

PWM frequency

The PWM frequency can be changed in object 0x2010:9 Torque controller: Switching frequency.

- Enter the value 0 for 16 kHz
- Enter the value 1 for 32 kHz
- Enter the value 2 for 64 kHz

By default, the switching frequency is set to 16 kHz. Higher switching frequency values will result in higher switching losses in the inverter and less copper and iron losses in the motor. Moreover, higher switching frequencies will result in less ripples in currents and voltages of the DC link which results in less ripple amplitudes in the currents and voltages of the power supply.

> Note:
>
> If the motor time constant is bigger than 100 ms, it’s recommended to use 16 kHz. If the

> motor time constant is less than 100 ms, or the ripple amplitudes of the DC-Link is too high, it's recommended to select higher switching frequencies.

Gear Ratio

A gear ratio between motor and driving shafts can be configured. This ratio defines the relationship between the motor shaft and driving output shaft. The position feedback and command will always be relative to the driving shaft.

$$\text{gear ratio} = \frac{\text{motor shaft revolutions}}{\text{driving shaft revolutions}}$$

The feature dual-loop cascaded position control can be used in such setups.

Advanced configurations

Motor Overload Protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating.

Cogging Torque Compensation Removes ripples caused by magnetic interaction between motor rotor and stator.

Field Weakening The velocity range of electric motors can be extended by weakening the magnetic field of the rotor linearly over the speed range.

Parameters

Object ID	Name	Description
0x2003	Motor Specific Settings	Parameters according to your motor specification
0x2010	Torque controller	Values Damping Ration and Settling time for tuning the current controller and switching frequency (PWM)
0x6075	Motor Rated Current	The RMS value of motor phase currents which can continuously pass through motor windings without overheating the motor
0x6076	Motor Rated	The amplitude of generated torque at the rated motor

Object ID	Name	Description
	Torque	phase currents
0x6080	Max motor speed	The maximum possible speed of the Motor according to the data sheet
0x6091	Gear Ratio	Object for indicating the revolutions of motor and shaft

Generic brake output voltage

Overview

The voltage output of the brake can be controlled manually in real-time using the ‘manual output voltage’ mode of the brake release object.

> Note:
> The brake release object can be used for either controlling a brake or a generic voltage but not both.

This allows for custom brake release strategies on the master, as well as providing a generic output for a single phase device connected to this phase (e.g. a fan).

This voltage is applied at:

- Phase D of SOMANET Node
- SBC out of SOMANET Node Safety

> Attention:
> The brake output can be **switched off safely** when SOMANET Node Safety is used but **not controlled safely**.

Usage

- Connect the device to the dedicated brake output of the drive and to Ground (GND).
- Set object ‘Brake options: Release strategy’ (subitem 0x2004:4) to ‘Manual output voltage’ (0), it is recommended to do so in ‘Brake Configuration’ of the Setup Wizard of OBLAC Drives.
- Use the SDO object ‘Brake options: Brake status’ (subitem 0x2004:7) for switching the output on and off:

  1: off

  2: on
- In the master assign ‘Brake options: Output voltage’ (subitem 0x2004:10) to PDO for controlling the output voltage in real-time, the unit is given in mV.
- While the master is running, control the output voltage by writing the value in mV to ‘Output voltage’.
- To turn off the output, write a 0 to ‘Output voltage’

> Note:
>
> The SBC signal has to provided as well when the voltage shall be switched on with Node Safety

Limitations

- Please note that the brake phase voltage control comes with limited accuracy.
- Use this feature only in low precision applications (e.g. load-dependant speed of a fan).

Parameters

Object ID	Name	Description
0x2004	Brake Release	• Subitem 0x2004:4 (Release strategy) can be used to activate the voltage on the brake output • Subitem 0x2004:7 (Brake status) turns it On and Off • Subitem 0x2004:10 (Output voltage) sets the voltage.

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

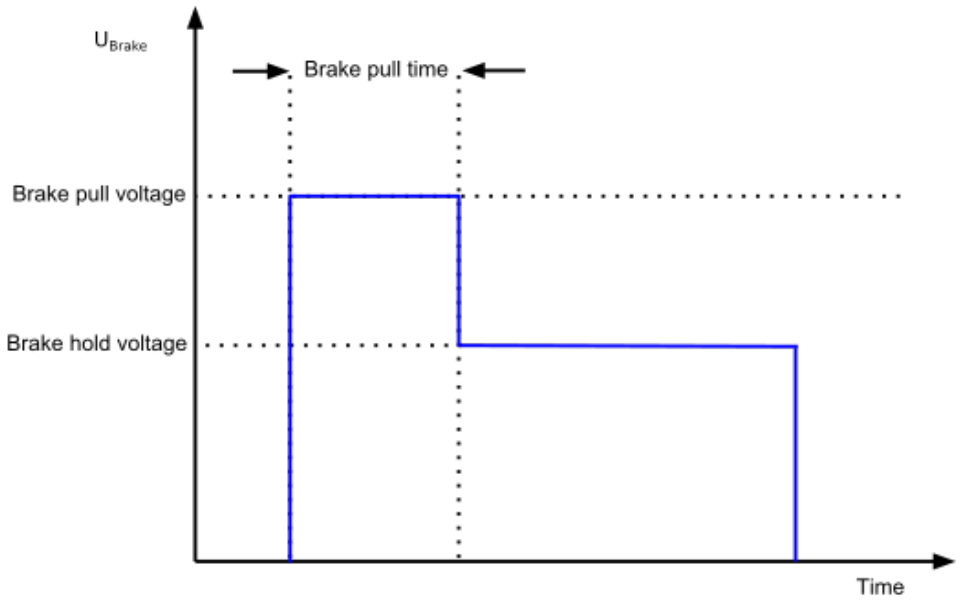
Brake configuration

In some cases, the motor is equipped with a mechanical brake which can be released by applying electrical voltage to the brake's terminals.

Detailed description

The following diagram shows the steps for the release procedure:

A configurable voltage is applied to the positive terminal of motor brake (It is assumed that the negative terminal of the brake is connected to Ground).



When the motion system has entered operation enabled, the specified "Brake pull voltage" will be automatically applied for a configurable amount of time designated as "Brake pull time". After this time period, the second voltage level "Brake hold voltage" will be applied to the brake terminals permanently. The brake hold voltage can be set to a lower value to save energy.

> Attention:
>
> Don't set the hold voltage too low or you risk entering a marginally stable situation where any jerk can engage the brake.

Once the motion system leaves the state "operation enabled" the voltage is automatically dropped to Ground after the designated time "Brake engage delay" and the brake is mechanically activated.

Pin-brake stategy

A pin-brake is a special type of brake commonly used in light-weight robot joints. The brake usually consists of a solenoid linear actuator fixed to robot body and a notched wheel fixed to the rotating motor shaft.

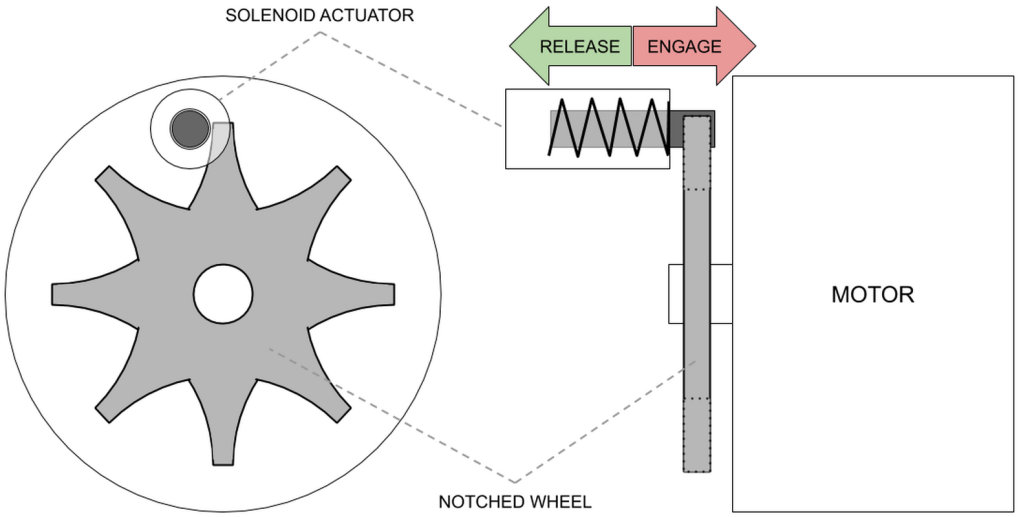


Image 1 - Pin brake (illustration exemplary)

To release a pin-type brake, the motor must move slightly so that the pin can slide out. This is mainly required with hanging loads (e.g. robot arm) where gravity is producing a torque pressing the wheel notch against the actuator pin. The following method is implemented to ensure reliable pin-type brake operation.

Releasing Pin brake in Different OP Mode

Diagnostic Mode

In Diagnostics Mode, the brake is controlled by the master itself using the `Brake Status` object (0x2004:7). So putting the drive in OP ENABLED state in diagnostics mode will not start the Pin brake release process. Brake will remain in the engaged position. But if the drive has a relative encoder with Index and QEI index is not known then enabling the drive in diagnostics mode will only internally trigger the Temporary offset detection. Brake will remain in the engaged position and no process to release the Pin brake will be initiated.

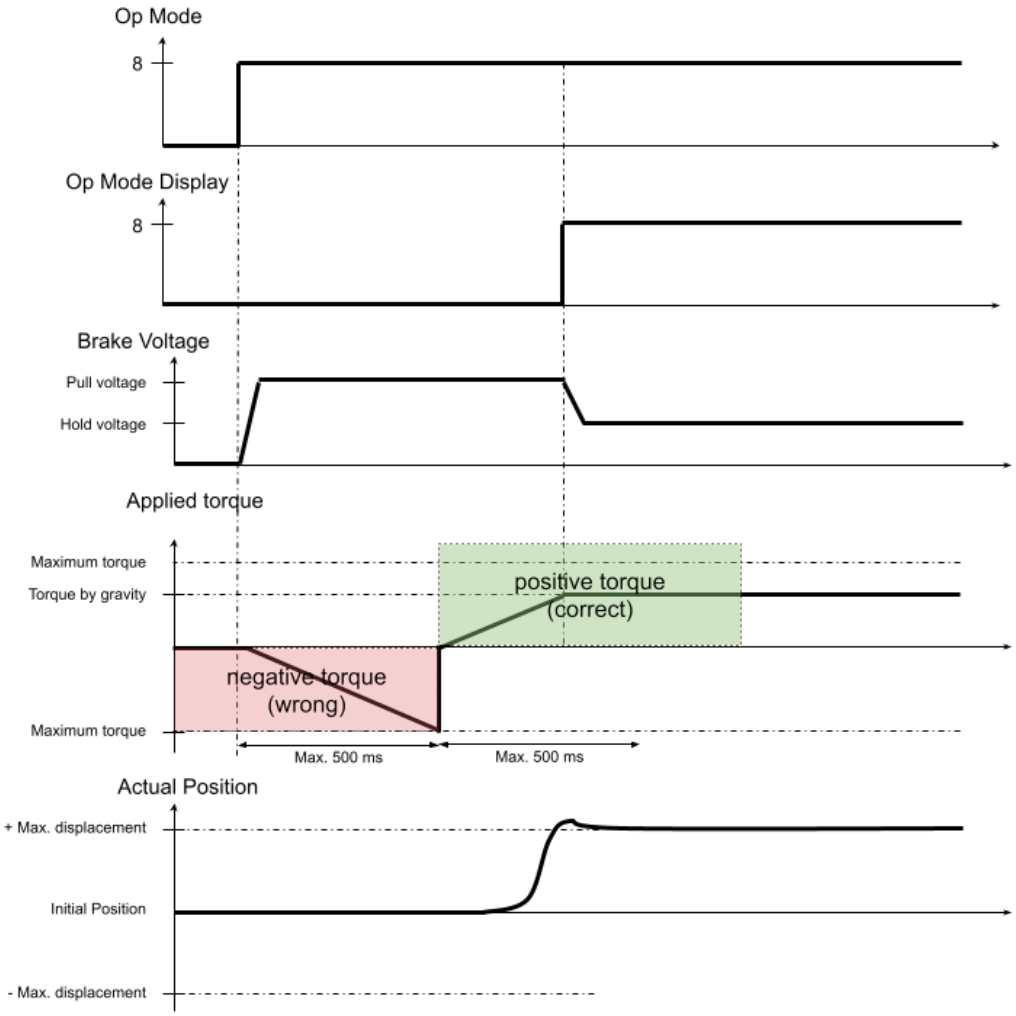
After the drive has been enabled, changing the brake status object to `DISENGAGE` will start releasing the pin brake.

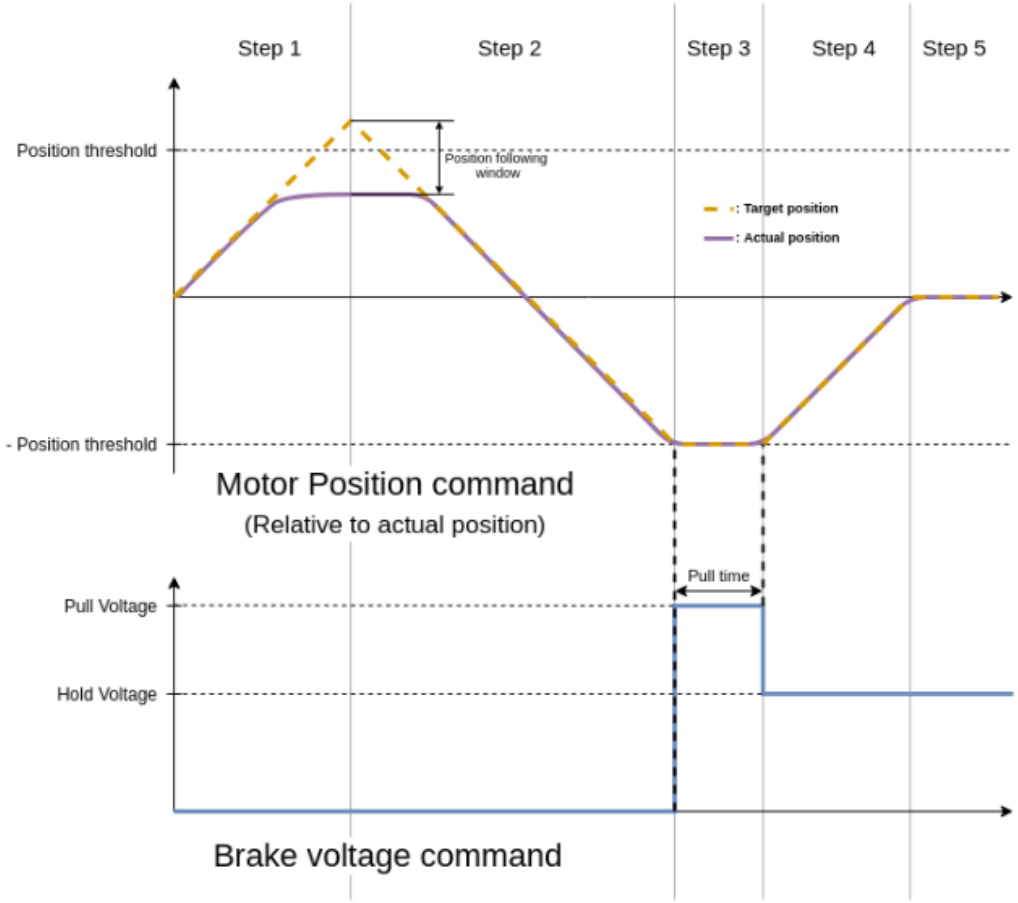
Manual Release using Brake Status Object

Controlling the brake using the brake status object 0x2004:7 will trigger the brake release procedure only if the drive is in Diagnostic mode and OP ENABLED state.

If the drive is not in OP ENABLED state, then releasing the brake will only apply voltage on PIN (Phase D).

Timing diagram





Step 0

In this step, the variables related to the pin brake release are initialized. The motion polarity is also determined based on the phase's connection (object 0x2003:05). The open loop profiler is configured.

Step 1

The open loop profiler moves the motor in positive / negative direction, depending on the polarity.

Step 2

The motor´s movement is blocked in any direction and it starts moving in the opposite direction.

Step 3

If the motor has reached the target position, the pull voltage is applied for the duration specified by pull time. During the application of pull time, the motor does not move.

Step 4

Target position is set to the initial position of the rotor and the open loop profiler motor moves to the initial position.

Step 5

Control is enabled, the drive accepts Position / Velocity / Torque targets.

The brake controller raises the torque progressively until it reaches the maximum displacement threshold. If it reaches the maximum torque threshold first, it rises the torque again in the opposite direction. When the torque is high enough to compensate for gravity, the load is lifted until the position reaches the threshold value, the torque stays to a holding torque value and the brake will be disengaged. After the release delay, the brake voltage is driven to “Brake hold voltage” and Op Mode display is set to Op Mode. The controller is then enabled. To use the pin-brake mode please select the correct “Release strategy” in the brake configuration section in OBLAC Drives commissioning software, or via direct object dictionary access.

> Note:
>
> The brake release is an open loop system. After the brake pull time it is assumed that releasing was successful and the hold voltage is applied.

Brake PWM switching frequency

The PWM switching frequency of the brake can be changed in object 0x2004:11 Brake options: Switching frequency.

- Enter the value 0 for 16 kHz
- Enter the value 1 for 32 kHz
- Enter the value 2 for 64 kHz

Custom brake modes

It is also possible to implement custom brake modes. For information please refer to Generic brake output voltage.

Parameters

Object ID	Name	Description
0x2004	Brake Release	All options that are required for brake release.
0x2004:8	Position Threshold Value	Same as current implementation
0x2004:9	Percentage of rated Current (pin brake)	Can be used to control the field rotational speed while releasing Pin brake

User defined velocity units

Overview

The *default unit* for all velocity and speed data is **rpm**

The *default unit* for all acceleration and deceleration data is **rpm/s**

In the case that the application requires a higher precision in velocity, the unit can be configured by the user through object 0x60A9 SI unit velocity.

Supported values are:

- 1 rpm (default value)
- 0.1 rpm
- 0.01 rpm
- 0.001 rpm

> Attention:
> As soon as this object is changed, **all related objects and subitems** also use the chosen unit and **must be scaled accordingly**. Unexpected behavior and accidents can occur if the chosen unit is not taken into account. For more information please refer to “Objects affected by changes of the velocity unit”.

> Attention:
> The velocity reference frame is reported on the moving shaft. The target and actual velocity are scaled with a factor of “gear-ratio”. For better precision, it is recommended to change the SI unit to mrpm.

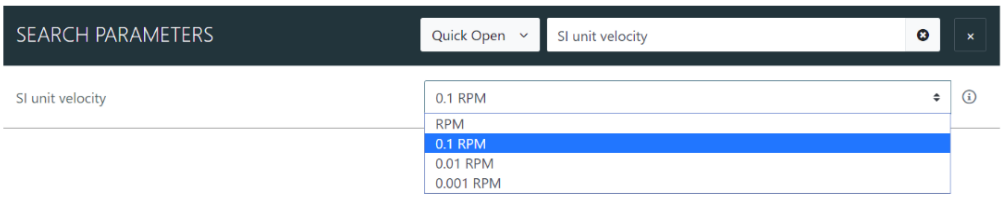
Usage

To configure the velocity unit write the following values through SDO to object 0x60A9 SI unit velocity:

- 1 rpm (value 0x00B44700 or 0d11814656) - default
- 0.1 rpm (value 0xFFB44700 or 0d4290004736)

- 0.01 rpm (value 0xFEB44700 or 0d4273227520)
- 0.001 rpm (value 0xFDB44700 or 0d4256450304)

Since version 19.1.0 of OBLAC Box this can be done directly in the UI using Quick Open:



After changing the unit please change all speeds /acceleration objects according to the new precision (see section below for details).

Click on “save to device” to keep the configuration.

Objects affected

- 0x606B Velocity demand value
- 0x606C Velocity actual value
- 0x606D Velocity window
- 0x606F Velocity threshold
- 0x6080 Max motor speed
- 0x6081 Profile velocity
- 0x6083 Profile acceleration
- 0x6084 Profile deceleration
- 0x6085 Quick stop deceleration
- 0x609A Homing acceleration
- 0x60B1 Velocity offset
- 0x60FA Control effort
- 0x60FF Target velocity

These subitems are also affected by the value:

- 0x6099:1 Homing speeds: Speed during search for switch
- 0x6099:2 Homing speeds: Speed during search for zero

- 0x2010:6 Torque controller: Field weakening starting speed
- 0x2010:7 Torque controller: Field weakening ending speed
- 0x2012:4 Position controller: Position loop integral limit
- 0x2013:6 Gain scheduling: Position loop integral limit 0
- 0x2013:15 Gain scheduling: Position loop integral limit 1
- 0x2013:2 Gain scheduling: Threshold velocity 0
- 0x2013:11 Gain scheduling: Threshold velocity 1

Example

When a dual-sensor setup is used with one encoder mounted after a reducer with a gear ratio of 1:100 the default velocity unit rpm causes severe inaccuracy in the motion control.

1. In OBLAC Box choose 0.001 RPM for SI unit velocity.

   When using a version of OBLAC Box before 19.1.0, manually write 0d4256450304 to object 0x60A9 SI unit velocity through SDO. The terminal of OBLAC Box can be used for this operation with the following code line:

   ```
   EtherCAT download -p0 0x60A9 4256450304
   ```

2. OBLAC Box will automatically scale the affected objects.

Parameters

Object ID	Name	Description
0x60A9	SI unit velocity	This object defines the velocity unit.
0x606B	Velocity demand value	Output of position controller in position mode/trajectory generator in velocity mode.
0x606C	Velocity actual value	Provides the value read at the velocity sensor (or the derivation of the position signal).
0x606D	Velocity window	This value reflects each side of the target velocity.
0x606F	Velocity	Configures the velocity that is considered zero

Object ID	Name	Description
	threshold	velocity.
0x6080	Max motor speed	Sets the maximum allowed speed for the motor in either direction.
0x6081	Profile velocity	Configures the velocity target for the end of the acceleration ramp during profiled motion.
0x6083	Profile acceleration	Configures the acceleration used during profiled motion.
0x6084	Profile deceleration	Configures the deceleration used during profiled motion.
0x6085	Quick stop deceleration	Deceleration used for stopping the motor when quick stop function is activated.
0x609A	Homing acceleration	Configures the acceleration and deceleration to be used during a homing operation.
0x60B1	Velocity offset	Sets a feed-forward velocity as input to the velocity controller.
0x60FA	Control effort	Provides a measure of the control effort (the output of the position control loop).
0x60FF	Target velocity	Sets the target velocity and will be used as input to the trajectory generator.
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)
0x2013	Gain Scheduling	Parameters needed for Gain Scheduling

Commutation Offset Detection

Overview

Controlling a motor with Field Oriented Control requires precise measurement of the motor's electrical angle. With phase order and commutation angle offset detection (hereinafter referred to as "Offset Detection"), the relationship between the motor electrical angle and the motor mechanical angle (position sensor feedback) is identified, afterwards the motor position sensor feedback is used to determine the motor's electrical angle for motion control.

> Note:
>
> This feature is needed for every motor control setup. The application will not work properly before the Offset Detection routine is successfully executed.
>
> If an incremental encoder without index is used for commutation, the offset detection has to be done on every power on. Otherwise, the routine only needs to be done once during the first commissioning and after mechanical changes of the motor or of the commutation encoder.

Basic Principles

The Offset Detection routines identify two parameters for the relationship between the motor electrical angle (rotor magnetic field angle) and the motor position sensor feedback:

1. The motor phase configuration (NORMAL or INVERTED) (0x2003:5)

Normally, both rotor angle and sensor angle should increase/decrease at the same time. This situation is called "NORMAL motor phase configuration" and is shown in Image 1.

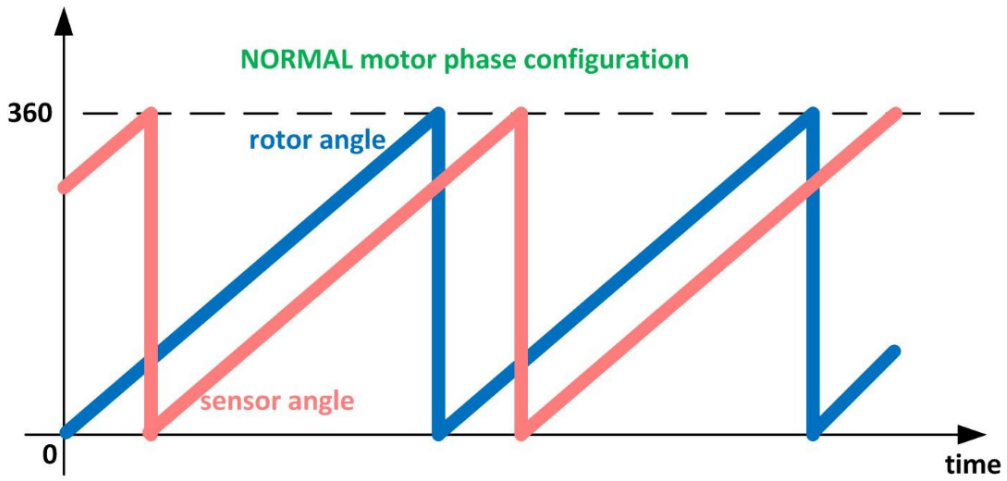


Image 1 - Normal motor phase configuration (sensor angle and rotor angle are both increasing/decreasing simultaneously)

However, sometimes the angle of position sensor is not increasing in the same direction of rotor angle. This can happen if the position sensor is mechanically flipped, or if two of the stator phases are flipped. Image 2 shows such a situation where the rotor angle and the position angle are not increasing/decreasing in the same direction. This situation is called *Inverted motor phase configuration*.

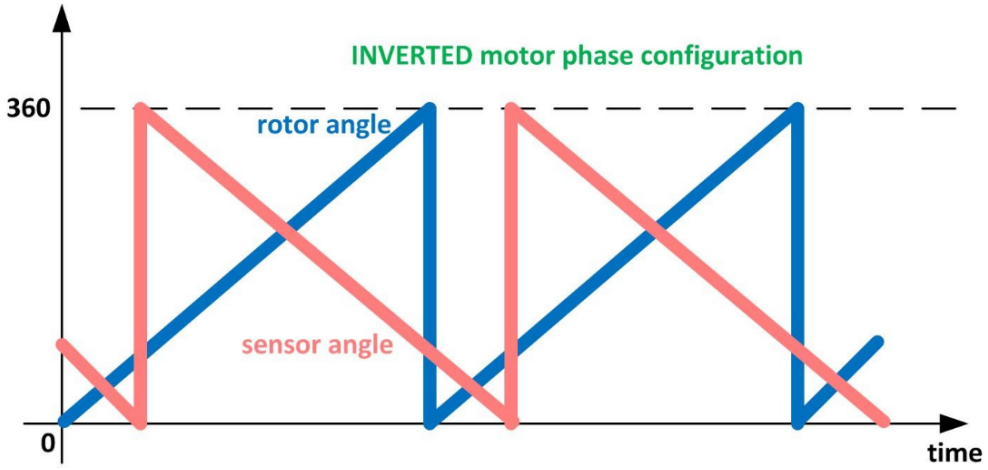


Image 2 - Rotor angle and position sensor angle are not increasing/decreasing in the same direction (Inverted motor phases)

2. The commutation angle offset (0x2001)

In most cases, the provided sensor angle contains a shift respect rotor angle. Image 3 shows such a case where the provided sensor angle has an offset respect rotor angle:

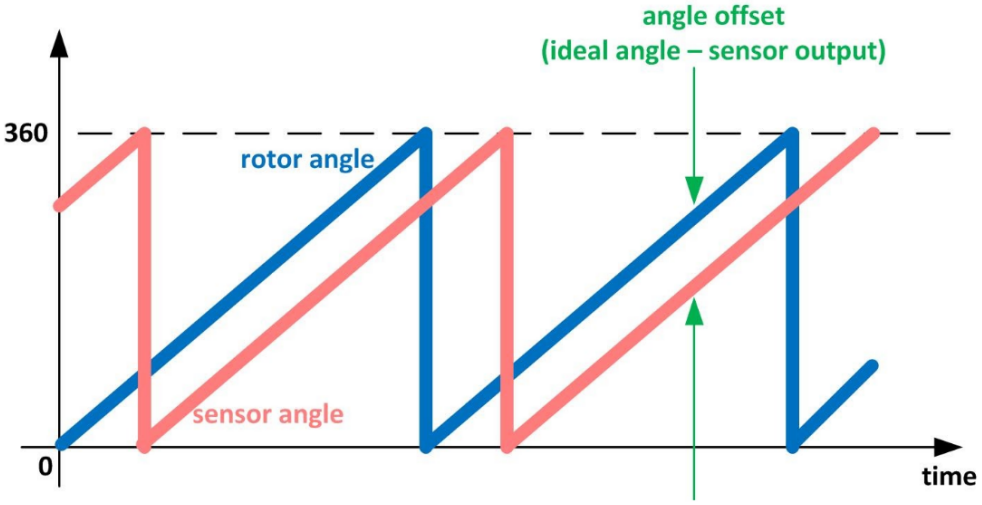


Image 3 - In most of the cases, the provided sensor angle has a commutation angle offset

Offset Detection is carried out in the Diagnostics Opmode using OS-Commands.

Note:

When quick stop is commanded during Offset detection, the procedure is aborted and the drive is disabled.

Requirements

Load requirements

- The active load of the system should be lower than motor rated torque
- The position feedback sensor on the motor side should be configured properly and provide proper feedback of the motor position
- The motor should be configured correctly
- The mechanical system should not block the motor
- The brake should be configured correctly and work properly during the routine so that it doesn’t block the system
- The power supply used should be able to provide the voltage / current required by the routine

Note:

If the requirements are not met, the Offset Detection routine fails and the detection result can be wrong, leading to further failures.

Note:

Large single side loads during the routine (such as the gravitational load) can lead to lower detection accuracy with Offset Detection method 0 and 1.

Note:

The precision of the commutation angle offset detection depends on the resolution of the

encoder that is used. For example, if HALL sensors with a resolution equal to 60 degrees are used, the precision of commutation angle offset detection is expected to be 60 degrees as well.

Autophasing methods

Three different methods for finding the “commutation angle offset” are implemented. The method can be selected manually (in subitem 0x2009:3). The optimal method is determined by the setup and the use case.

Method 0

Method 0 is the option used by default. This algorithm has the following characteristics:

- It needs no tuning
- It is able to hold the load
- The offset can be precisely measured
- It will rotate the rotor up to 100% of a pole-pair range

Method 1

Attention:

This method is still in prototype phase and can be tested at the user’s own risk.

This algorithm has the following characteristics:

- It needs previous tuning of the controller with a dedicated set of gains: the phasing controller whose parameters are located in object 0x2009
- The offset can be found with a limited movement range. With a well tuned PID controller the movement is typically less than 5 degrees
- The offset can be measured very precisely

Method 2

> Attention:
>
> This method is still in prototype phase and can be tested at the user's own risk.

This algorithm has the following characteristics:

- It needs no movement (no rotation)
- If a brake is available, the brake should be engaged while offset detection method 2 is in progress.

> Attention:
>
> If no brake is available and mechanical load is relatively small (compared to rated motor torque), then offset detection method 2 can lead to rotor movement

- It is not able to hold the load. As a result, **when a brake is present, the brake will remain engaged during the procedure of offset detection method 2.**
- It can take up to 250 milliseconds.
- The precision of method 2 is not as high as methods 0 or 1.

> Note:
>
> This method can cause audible noise when it's executed.

Comparison of methods

The optimal method depends on the use-case. The following tables shows the advantages and disadvantages of each method.

Method #	Tuning necessary?	Able to hold a load?	Precision of measurement	Rotation required	Notes
0	No	Yes	Good	up to 1 pole-pair range	
1	Yes	Yes	Good	Depends on quality of tuning. Typically less than 20 elec-trical degrees.	
2	No	No*	Less good	None	Duration up to 250 ms Can cause audible noise

* This procedure is usually carried out with applied brakes.

Common Use Cases

Motion is allowed

- Method 0/1 should be used
- Method 0 gives perfect precision without any previous tuning.
- If the final system has coupled joints or heavy or single side loads it might be required to perform Offset detection before assembly for optimal performance.

- If the movement of up to one pole pair is not acceptable method 1 can be selected, where the movement can be much less. The PID controller used for method 1 needs to be tuned carefully.

> Note:
> Try moving the arm to a position in which gravity has no or little impact, for optimal performance of the detection. For example the center of gravity should be right above or right below the axis of motion.

Motion is not allowed

When no movement is allowed during commissioning, method 2 can be used. This may happen when the setup is already in the application. Note that the method has less detection accuracy, thus the commutation efficiency can be less. In this case, it's recommended to redo Offset Detection when the setup is allowed to move.

Special case: Offset detection Method 2 with an Incremental encoder with index

If an ABI encoder is used for commutation, a routine called "finding index" is usually required when enabling the drive (the first time after booting the system), which will rotate the motor to find the index of the sensor. With commutation method 2, the drive can find an internal temporary offset and use this offset until the index is detected. Once the index is detected, it replaces the temporary offset by its corresponding value in the object dictionary. In practice, Offset Detection can be carried out with method 0/1 during commissioning, the result can be saved and method 2 can be executed. This way, no movement is required when enabling the setup, and the torque control performance of the setup is still promised.

Usage

Offset detection is usually carried out in the setup section of OBLAC Box. The commutation offset detection method is chosen with subitem 0x2009:3. The default method is 1.

> Attention:
> When the existing Offset Detection results become invalid, try locating the cause and fix the system-level failure instead of running Offset Detection again. Else further failures can happen.

The procedure can also be triggered manually for production use:

Steps to activate the “commutation angle offset chain”:

- In object 0x6060: (Modes of operation) set the value to -2 (Diagnostics mode)
- In subitem 0x2009:3 select the offset detection method (values: 0, 1 or 2)
- In object 0x6040 Controlword switch the CiA402 status to “Operation_Enabled”
- The procedure will start automatically.
- Trigger the following OS commands to carry out offset detection:
    - Command 6: Open phase detection
    - Command 8: Phase resistance measurement
    - Command 9: Phase inductance measurement
    - Command 7: Pole pair detection
    - Command 4: Motor phase order detection
    - Command 5: Commutation offset measurement

Autophasing during initial configuration of a drive with absolute encoder:

The following steps are mandatory:

- Command 6: Open phase detection
- Command 7: Pole pair detection (on new type of motor)
- Command 4: Motor phase order detection
- Command 5: Commutation offset measurement

Autophasing after power on with incremental encoder:

The following steps are mandatory:

- Command 4: Motor phase order detection
- Command 5: Commutation offset measurement

> Note:
>
> Motor phase order detection is mandatory in order not to damage the system.

Tuning gains (method 1)

The gains for PID tuning can be found in object 0x2009:

- Subitem 0x2009:4 defines K_p for method 1.
- Subitem 0x2009:5 defines K_i for method 1.
- Subitem 0x2009:6 defines K_d for method 1.

> Note:
>
> As a rule of thumb, the following approach can be applied:
>
> Start values of the gains:
>
> $K_p = 1$
>
> $K_i = 1$
>
> $K_d = 0$
>
> Increasing and decreasing K_p and K_i will have the same effect as increasing them in a simple PID position controller.
>
> If needed, the increase in K_d should be done in very small steps (such as changing from 0 to 0.001), this will reduce the oscillations of the load.

Parameters

Object ID	Name	Description
0x2001	Commutation angle offset	Represents the required shift for position sensor output.
0x2003	Motor Specific Settings	Details for your motor specification.
0x2009	Commutation offset	• Subitem 0x2009:1 shows the state of the commutation offset. • Subitem 0x2009:2 defines the amount of torque that is applied during the offset detection process.

Object ID	Name	Description
		• Subitem 0x2009:3 determines the method to be used for offset detection. • Subitem 0x2009:4 defines K_p for method 1. • Subitem 0x2009:5 defines K_i for method 1. • Subitem 0x2009:6 defines K_d for method 1.
0x6075	Motor Rated Current	Determines what percent of rated motor currents should be injected into the motor phases during the offset detection procedure.
0x6076	Motor Rated Torque	The rated torque of your Motor according to the data sheet.
0x6060	Modes of operation	Selects the mode of operation.
0x6040	Controlword	Sets the status of the drive

Power Limit

Overview

In case the DC power supply of the Drive has limited output power (for example limited output I_{DC} at a certain V_{DC} level) this feature can be used to make sure that the output power drawn from the power supply is limited.

Use case:

Battery powered system

A good example of such cases are applications where batteries are used, and their output currents should remain below a certain limit.

This feature does not measure the mechanical torque directly. Instead it is using the “computed electrical torque” and assumes that the “computed electrical torque” is equal to the

real generated mechanical torque of the motor.

It calculates the rotational speed of the motor with values from the position sensor at the motor shaft.

Requirements

This feature works in CST, CSV, CSP, PT, PV and PP modes.

The following parameters must be properly set from the motor data sheet:

- Motor torque constant
- Position sensor configurations
- Number of pole pairs

Detailed Description

Mechanical output power of the Drive in any condition is equal to:

$$P_{out}\left[W\right] = \tau\left[Nm\right] \cdot \omega\left[\frac{rad}{s}\right]$$

where τ represents Torque and ω represents the rotational speed.

At any operational point, the Drive has access to rotational speed. As a result the limit of generated torque can be calculated as:

$$\tau_{limit}\left[Nm\right] = \frac{P_{out,max}\left[W\right]}{\omega\left[\frac{rad}{s}\right]}$$

This limit is updated in each execution loop of the torque control, and it limits the absolute value of reference torque.

Note:

By default the power of each Drive is internally limited to 110% of its maximum peak output power. As a result, this object will only be effective if its value is less than 110% of Drive's maximum peak output power.

Parameters

Object ID	Name	Description
0x200B	Max power	Sets the limit in Watt
0x203F	Error Report Object	Shows the warning “MxPwrLmt” once the limit is reached

Encoders and I/O

Besides various standard interfaces for absolute and incremental encoders, SOMANET devices are equipped with a multitude of general purpose digital and analog IOs. These can be both directly propagated to a higher level controller via the field-bus interface, or configured to internally be used for triggering, homing, etc.

It’s also possible to connect devices such as valves, magnetic actuators or LEDs to digital outputs.

For attaching a brake, please refer to brake configuration.

Analog In 163
Analog Input Scaling 165
Digital I/O 168
Digital GPIO 169
Encoder Configuration 180
Encoder sampling frequency 182
Commutation with Hall and ABI combined 183
Input counter 184

Analog In

Analog sensors usually output a voltage or a resistance that can be interpreted by the firmware. When the output is a voltage, the signal can be directly read by the analog input, but if it’s a resistance (e.g. resistance temperature detectors), a voltage divider must be used to generate a voltage equivalent to the resistance.

Important:

If your servo drive does not incorporate a voltage divider itself, you must attach one separately.

SOMANET servo drives come with a default configuration for the analog inputs that can be alternated on request.

For details please refer to SOMANET Node "Analog In specification" .

Using single-ended sensors on differential inputs

A single-ended encoder can be attached to the differential ports, if the voltage range fits only to the differential inputs. The signal must be connected to the +pin of the Analog In and the Ground to the -pin.

Note:

In this case, half of the analog input range will be lost.

Nonlinear conversion

- Analog Input Scaling: Nonlinear conversion of an analog voltage using a 5th order polynomial function.

Parameters

Object ID	Name	Description
0x2401	Analog input 1	Parameter holding the value of analog input 1
0x2402	Analog input 2	Parameter holding the value of analog input 2
0x2403	Analog input 3	Parameter holding the value of analog input 3
0x2404	Analog input 4	Parameter holding the value of analog input 4

Analog Input Scaling

The analog input value is converted into digital ticks from 0 (representing U_{min} of the selected input, e.g. -5 V) to 65535 (representing U_{max} of the selected input, e.g. +5 V) which must be computed externally for further use.

This feature allows a nonlinear conversion of one of the available analog inputs using a polynomial function up to 5th order and to read the value directly.

It's also possible to describe a voltage divider connected to the analog input for sensors that output a resistance as measurement value.

> Use case:
>
> Temperature sensor: Applications include converting the output of a temperature sensor directly into degrees Celsius, Fahrenheit or any other unit desired by the user.
>
> For details about using temperature sensors check the chapter Temperature sensors in the System integration guide.

> Use case:
>
> Torque sensor: Applications include converting the output of a torque sensor directly into mNm, Nm or any other unit desired by the user.

Requirements

- The hardware must support at least one analog input

Usage

Wiring

Follow this wiring scheme when attaching an external voltage divider.

Software configuration

The object used for this feature is 0x2038 Analog Input Scaling. The converted value will be reflected in Subitem 0x2038:1.

The polynomial used for the conversion can be described with the parameters "Constant a0" (0x2038:4) to "Constant a5" (0x2038:9), which are the coefficients for the polynomial function.

The analog input to be used can be defined using the parameter “Analog input” in subitem 0x2038:2.

This parameter has 4 possible options, which are the standard analog inputs (values: 1-4).

- Analog input selection k (0x2038:2): Select the scaled measurement source. Values:
  - 1 = Analog Input 1
  - 2 = Analog Input 2
  - 3 = Analog Input 3
  - 4 = Analog Input 4

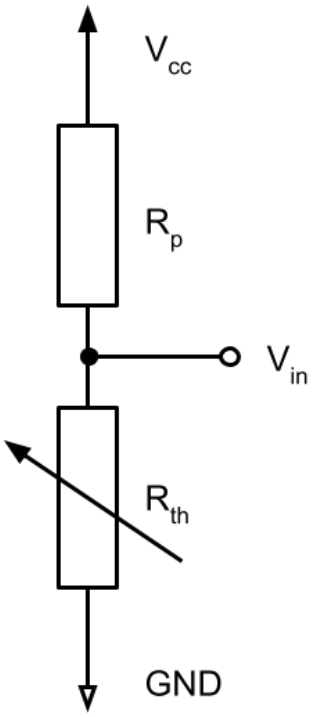
The unscaled raw analog input values are filtered through a first order low pass filter before they are scaled. The cut-off frequency of this low-pass filter can be configured in subitem 0x2038:10.

> Note:
>
> The external scaled measurement feature is using the filtered analog input value. Therefore all the analog inputs mentioned in this section are referring to the filtered analog input.

A 5th order polynomial function with or without voltage-divider is supported.

This feature is also designed to have a different behavior depending on the value of the parameter “Resistance” (0x2038:3). If this value is set to 0, the polynomial will be directly applied to the value of the ADC count, which has a range of 0-65535 (0 to V_{cc}).

If a voltage divider is connected to the analog input, as shown in the figure below, the value of the “Resistance” parameter (0x2038:3) defines the value of the pull-up resistor R_p of this voltage divider.



- V_{cc} represents the value of voltage at which the ADC is connected to, causing the ADC to count an output of 65535.
- R_p represents the pull-up resistor.
- R_{th} is a resistive sensor, e.g. a thermistor or strain gauge.
- V_{in} is the voltage that is connected to the selected analog input.

The cut-off frequency of the low-pass filter (0x2038:10) for the analog input can also be tweaked to further fine-tune the unscaled analog input before feeding it to the scaling algorithm.

Thresholds

Upper and lower level thresholds can be configured in subitems 0x2038:11 and 0x2038:12. When these thresholds are violated, the error **ExAnSnsr** is triggered in the Error report object.

> Note:
>
> When the value approaches 95% of the defined thresholds, a warning is triggered.

Polynomial

If R_p := 0 (no voltage divider)

$$y(u_k) = a_5 * u_k^5 + a_4 * u_k^4 + a_3 * u_k^3 + a_2 * u_k^2 + a_1 * u_k + a_0$$

If $R_p > 0$ (user specified pull-up resistor for voltage divider)

$$y(u_k) = a_5 * R_{th}^5 + a_4 * R_{th}^4 + a_3 * R_{th}^3 + a_2 * R_{th}^2 + a_1 * R_{th} + a_0$$

with

$$R_{th} = \frac{u_k}{4096} * \frac{R_P}{1 - \frac{u_k}{4096}}$$

The coefficients and the pull-up resistor values are specified in object 0x2038, subitems 2-9

- 0x2038:3 (Resistance R): Select the pull-up resistor connected to the selected analog input. The value is given in Ohm.
- 0x2038:4 (Constant a0): Constant a0 of the polynomial (Offset).
- 0x2038:5 (Constant a1): Constant a1 of the polynomial (1st order coefficient)
- 0x2038:6 (Constant a2): Constant a2 of the polynomial (2nd order coefficient)
- 0x2038:7 (Constant a3): Constant a3 of the polynomial (3rd order coefficient)
- 0x2038:8 (Constant a4): Constant a4 of the polynomial (4th order coefficient)
- 0x2038:9 (Constant a5): Constant a5 of the polynomial (5th order coefficient)

Parameters

Object ID	Name	Description
0x2038	External scaled measurement	Transforms an analog input to an output by a 5th order polynomial

Digital I/O

SOMANET supports several Digital I/O which can be either used as GPIO or SPI SSI (single-ended), I²C, BiSS (single-ended).

Details and default settings are described in detail for Encoders and IO.

Configuration of Digital I/Os as Input or Output

Most Digital I/Os are configurable as either input or output independently if they are used as general purpose input/output (GPIO). This is done in 0x2210 GPIO pin configuration.

Subitems 1-4 describe the available pins and can be configured with these values:

- 0 Disabled (default)
- 1 Input without pull-down
- 2 Input active high - a pull-down resistor will be activated
- 3 Output

Using Digital I/Os for input actions or output events

It's possible to use the Digital Inputs of the Drive to command actions and to configure the Digital Outputs to trigger when certain events happen. This can be used to control some aspects of the Drive with an external signal or to control an external device with the Drive:

- Digital GPIO: Assigns actions to changes in the Digital Inputs and triggers Digital Outputs

Parameters

Object ID	Name	Description
0x2210	GPIO	Select if Digital I/Os are input or output

Digital GPIO

Overview

This feature allows using the Digital Inputs of the Drive to command actions and to configure the Digital Outputs to trigger when certain events happen. This can be used to control some aspects of the Drive with an external signal or to control an external device with the Drive.

Available features are:

Value	Definition	
0		Disabled
2	[IN]	General Purpose Input
3	[OUT]	General Purpose Output
4	[IN]	Home Switch Active High - Level Triggered
5	[IN]	Home Switch Active Low - Level Triggered

Value	Definition	
6	[IN]	Positive Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)
7	[IN]	Positive Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)
8	[IN]	Negative Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
9	[IN]	Negative Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
10	[IN]	Interlock (Disable Operation) Active Low - Level Triggered
11	[IN]	Fault Reset - Edge Triggered
12	[IN]	Interlock (Fault Reaction) Active Low - Level Triggered
13	[IN]	Time Stamped Rising Edge - Edge Triggered
14	[OUT]	Mimics Target Reached (tr) Bit of Statusword
15	[OUT]	Mimics Internal Limit Active (ila) Bit of Statusword
16	[OUT]	Timestamped Triggered Output
17	[OUT]	Position Triggered Output (Pulse)
18	[OUT]	Mimics Fault (f) Bit of Statusword

Value	Definition	
19	[OUT]	Mimics Switched On (so) Bit of Statusword
20	[OUT]	Mimics Ready to Switch On (rtso) Bit of Statusword
21	[OUT]	Mimics Operation Enabled (oe) Bit of Statusword
22	[OUT]	Mimics Voltage Enabled (ve) Bit of Statusword
23	[OUT]	Mimics Switch On Disabled (sod) Bit of Statusword
24	[OUT]	Mimics Quick Stop (qs) Bit of Statusword
25	[OUT]	Mimics Warning (w) Bit of Statusword

Usage

You can access the features in OBLAC Drives: Enter Feedback & IO/Digital IO

Manual configuration

The actions and events on GPIO pins can also be configured manually in object Object 0x2210 (GPIO Pin Configuration).

In this object a feature can be selected for each Digital I/O pin.

Some features require additional settings in other objects as well.

> Attention:
>
> Ensure that the selected pin allows the direction of the feature:
>
> An **input** action can only be configured on a Digital **In** pin.
>
> An **output** event can only be configured on a Digital **Out** pin.

Attention:

Each feature can only be used on one pin (except for General Purpose Input and General Purpose Output)

Description of features

General purpose In- and Output (GPIO)

General Purpose Input (2)

The high and low logic (voltage levels) at the selected GPIO pin will be reflected in the corresponding bits of object 0x60FD (Digital Inputs) as 1 and 0 respectively.

General Purpose Output (3)

The value of the selected digital output is high if the corresponding bit in sub-object 0x60FE:1 is high and the corresponding mask bit in sub-object 0x60FE:2 is set as 1.

The output value will be the value set in the object 0x60FE:1 (Digital Output).

Using Digital GPIO with Home switches and limit switches

Note:

Limit switch and Home switch configuration are used when the drive is in Homing Mode.

Limit switches can also trigger Quick Stop in all modes. In homing mode Quick Stop is triggered by the limit switch if it is not being used by the homing method.

Home Switch Active High - Level Triggered (4)

Active and inactive state of the Home switch will be reflected as 1 and 0 respectively at the selected digital input.

Home Switch Active Low - Level Triggered (5)

Active and inactive state of the Home switch will be reflected as 0 and 1 respectively at the selected digital input.

Note:

Irrespective of the switch type (Active High or Active Low), active and inactive states of the home switch will be shown as 1 and 0 respectively in bit 2 (home switch) of the Digital Input Object 0x60FD.

Positive Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct) (6)

Active and inactive state of the Positive limit switch will be reflected as 1 and 0 respectively at the selected digital input.

Positive Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct) (7)

Active and inactive state of the Positive limit switch will be reflected as 0 and 1 respectively at the selected digital input.

Note:

Irrespective of the switch type (Active High or Active Low), active and inactive states of the positive limit switch will be shown as 1 and 0 respectively in bit 1 (positive limit switch) of the Digital Input Object 0x60FD.

Note:

Positive Limit switch also triggers Quick Stop, if the drive is either in any opmode other than HOMING or the drive in is HOMING mode but the switch is not being used by the homing method. If Positive Limit Switch has been activated in this case, then bit 2 (qs) of the Controlword will be cleared and Quick Stop will be executed.

Negative Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct) (8)

Active and inactive state of the Negative limit switch will be reflected as 1 and 0 , in the corresponding bit (according to configured GPIO) of the object 0x60FD (Digital Inputs)

Negative Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct) (9)

Active and inactive state of the Negative limit switch will be reflected as 0 and 1 , in the corresponding bit (according to configured GPIO) of the object 0x60FD (Digital Inputs)

> Note:
> Irrespective of the switch type (Active High or Active Low) , active and inactive states of the negative limit switch will be shown as 1 and 0 respectively in bit 0 (negative limit switch) of the Digital Input Object 0x60FD.

> Note:
> Negative Limit switch also triggers Quick Stop, If drive is either in any opmode other than HOMING or drive in is HOMING mode but the switch is not being used by the homing method. If Negative Limit Switch has been activated in this case, then bit 2 (qs) of the Controlword will be cleared and Quick Stop will be executed.

Driving out of the limit switch is only possible in the opposite direction, the concrete behavior is mode-dependant:

Modes CSP, PP , PV , CSV , CST and PT

If a target is commanded which would move the axis further into the direction of the switch, the motor will stand still, since further movement in the direction of switch is not allowed.

Cogging torque compensation

If the switch is active and the user tries to switch to cogging torque compensation, this will not be allowed.

The error based on the activated switch will be raised by firmware.

Diagnostics mode

The following OS commands are not executed, if either of the limit switches is active:

- OS command 2 (Open Loop Field Mode)
- OS command 4 (Motor Phase Order Detection)
- OS command 5 (Commutation offset detection **with Methods 0 and 1**)
- OS command 6 (Open phase detection)
- OS command 7 (Pole pair detection)
- OS command 8 (Phase Resistance Measurement)
- OS command 9 (Phase Inductance Measurement)

If any of these OS commands is called while a limit switch is active, the response will be **OSCMD_ERROR_CMD_NOT_ALLOWED**.

The following OS commands can be executed, even if either of the limit switches is active:

- OS command 0 (Encoder register communication)
- OS command 1 (iC-MU calibration mode)
- OS command 3 (HRD streaming)
- OS command 5 (Commutation offset detection **with Method 2**)

Homing modes

If the switch is active then Homing is only allowed in those modes that will take the axis away from the activated switch. All those methods which try to move the axis further into the limit switch will raise an error.

The error raised depends on the limit switch being active:

- For a positive limit switch the error raised will be **PosLmAct**
- For a negative limit switch the error raised will be **NegLmAct**

Input actions

> Use case:
> Interlock: When connecting different devices for a common process it's often necessary to ensure the correct order of operation. This can be done by routing the signals to the master but it comes at a cost for reaction-time, data load and additional wiring. Input actions allow

nterlocking an operation directly at the Drive to guarantee the correct order of the process.

Interlock (Disable Operation) Active Low - Level Triggered (10)

Clear bit 3 (eo) of controlword when the selected digital input is low.

This will cause the drive to transition into Operation disable state.

Interlock will remain active as long as the configured GPIO remains low and transitioning into OP Enabled state will not be possible. Transitioning into OP Enabled state will only be possible when interlock is deactivated by setting the GPIO input high. Bit 3 (interlock bit) of the Digital Inputs Object 60FD will also be high when Interlock is active.

Fault reset (11)

Set bit 7 (fr) of controlword when the selected digital input is high.

Interlock (Fault Reaction) Active Low - Level Triggered (12)

Raises the fault “Interlck”, when the selected digital input is low.

This will cause the drive to transition into Fault Reaction State.

Interlock will remain active as long as the configured GPIO remains low and entering into OP Enabled state will not be possible. If OP Enabled state is attempted to be entered while interlock is active, *Interlck* fault will be raised again by the firmware. Transitioning into OP Enabled state will be possible when interlock is deactivated by setting the configured GPIO input high. Bit 3 (interlock bit) of the Digital Inputs Object 60FD will also be high when Interlock is active.

Time Stamped Rising Edge - Edge Triggered (13)

Report a rising edge timestamp with 2 µs accuracy if GPIO is a fast port (as indicated in the table below). Otherwise the accuracy is 62.5 µs. The value of this GPIO is updated in shared memory at 16 kHz.

Output events

Mimics Target Reached (tr) Bit of Statusword (14)

The value of the selected Digital Output mimics bit 10 of the Statusword (tr). When the “tr” bit is high, the configured digital output will be high

Mimics Internal Limit Active (ila) Bit of Statusword (15)

The value of the selected Digital Output mimics bit 11 of the Statusword (ila). When the “ila” bit is high, the digital output value will be high.

Timestamp triggered Output (16)

Trigger the Digital Output at a certain timestamp. The output value will be the value set in the object 0x60FE:1 (Digital Output).

The value specified in subitem 1 of object 0x2211 (GPIO output events) is the Timestamp at which the GPIO should trigger. The unit is in 10 ns.

The value in 0x2211:1 can be used for setting this timestamp. But the unit must be converted to 10 of ns.

> Note:
>
> The value need to be converted to be in the same time reference (1 µs vs. 10 ns)

> Use case:
>
> The value read in object 0x20F0:0 is 2385309 µs. The Timestamp triggered output shall be triggered 15000 µs after the current time. Both times need to be converted to the units of 0x2211:1 and added: 2385309 * 100 + 15000 * 100 = 240030900.

> Attention:
>
> When the DC Clock feature is not used, there is a drift between the timestamp and the DC Clock.

> Note:
>
> If the GPIO is a fast port, this is activated with an accuracy of 10 ns. Otherwise the accuracy is 62.5 µs.

Please refer to the table below for the type of port:

	type of port
Digital I/O 1	Fast port
Digital I/O 2	Fast port
Digital I/O 3	Fast port
Digital Output 4	N/A
Digital Input 7	N/A

Position Triggered Output (Pulse) (17)

Output a configurable width pulse to the selected Digital Output at a certain encoder position. The encoder port that is used can be selected in subitem 2 of object 0x2213:

	encoder port
Encoder Connector 1	Encoder Port 1
Encoder Connector 2	Encoder Port 2
Encoder Connector 3	Encoder Port 3 / Digital IO
Encoder Connector 4	
Encoder Connector 5	

> Note:
> If an ABI encoder is used, the response will be almost immediate (within 0.34 µs). If another encoder type is selected, the response will be slower (within 62.5 µs).

Object 0x2213 allows further configuration of this output event by setting the following parameters:

- Pulse Width
- Position Setpoint at which the pulse needs to occur at GPIO
- Selection of the encoder for which setpoint is defined.

Mimics Fault (f) Bit of Statusword (18)

The value of the selected Digital Output mimics bit 3 of the Statusword (f). When the “f” bit is high, the selected digital output will be high.

Mimics Switched On (so) Bit of Statusword (19)

The value of the selected Digital Output mimics bit 1 of the Statusword (so). When the “so” bit is high, the selected digital output will be high.

Mimics Ready to Switch On (rtso) Bit of Statusword (20)

The value of the selected Digital Output mimics bit 0 of the Statusword (rtso). When the “rtso” bit is high, the selected digital output will be high.

Mimics Operation Enabled (oe) Bit of Statusword (21)

The value of the selected Digital Output mimics bit 2 of the Statusword (oe). When the “oe” bit is high, the selected digital output will be high.

Mimics Voltage Enabled (ve) Bit of Statusword (22)

The value of the selected Digital Output mimics bit 4 of the Statusword (ve). When the “ve” bit is high, the selected digital output will be high.

Mimics Switch On Disabled (sod) Bit of Statusword (23)

The value of the selected Digital Output mimics bit 6 of the Statusword (sod). When the “sod” bit is high, the selected digital output will be high.

Mimics Quick Stop (qs) Bit of Statusword (24)

The value of the selected Digital Output mimics bit 5 of the Statusword (qs). When the “qs” bit is high, the selected digital output will be high.

Mimics Warning (w) Bit of Statusword (25)

The value of the selected Digital Output mimics bit 7 of the Statusword (w). When the “w” bit is high, the selected digital output will be high.

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

Parameters

Object ID	Name	Description
0x2211	GPIO output events	Timestamp at which the GPIO should trigger for “Timestamped Triggered Output”
0x2212	GPIO input actions	Time of the last rising edge for “Timestamped Rising Edge”
0x2213	GPIO position trigger	Sets Pulse Width, Position Setpoint and the encoder for “Position Triggered Output”

> Note:
>
> The pulse is generated independently of the direction from which the setpoint is crossed.

Object ID	Name	Description
0x60FE	Digital Outputs	Manufacture specific bits set the state of digital output
0x60FD	Digital Inputs	Manufacture specific bits show the state of digital inputs

Encoder Configuration

Parameter description

Function

Each encoder can have one of these functions: Commutation, Velocity or position.

It is of vital importance to have **exactly one** encoder configured for Commutation so that the motor shaft’s position is known to the commutation algorithm. In case your setup requires it, you can use a different encoder for Position or Velocity. In this case, the demanded and actual velocities will be scaled to the same reference frame of the position.

Resolution

For correct function it is necessary to provide the resolution of the encoder, this value is given in **Inc/Revolution**, please refer to your encoder's data sheet.

> Important:
>
> Please keep in mind that manufacturers sometimes give this specification in differing units.
>
> If the resolution according to your data sheet is not given in Inc/Revolution, please convert the value.

Zero velocity threshold

Time without a position change in which the velocity is assumed to be zero. After this time, the velocity value is set to 0 if no new position was received from the encoder. The default value is 5000 µs.

> Attention:
>
> Expert setting. The default value will perform sufficiently in most cases.

Polarity

The polarity represents the direction of rotation of the motor in comparison to the rotation of the encoder. If these directions don't match, then the polarity is inverted.

```
Set the value to -1 to invert the polarity.
```

Multiturn resolution

In case your encoder supports multiturn functionality, the number of stored revolutions must be specified here.

Parameters

Object ID	Name	Description
0x2110	Encoder 1 configuration	Configuration object for Encoder 1

Object ID	Name	Description
0x2111	Encoder 1 feedback	Provides the position or velocity value read at Encoder 1
0x2112	Encoder 2 configuration	Configuration object for Encoder 2
0x2113	Encoder 2 feedback	Provides the position or velocity value read at Encoder 2
0x2012:9	Position controller: Encoder source	Which encoder should be used for position control
0x2011:5	Velocity controller: Encoder source	Which encoder should be used for velocity control
0x2010:12	Torque controller: Encoder source	Which encoder should be used for torque control and for commutation

Encoder sampling frequency

Overview

The sampling frequency of BiSS, SSI or Nikon A-format encoders can be selected in the Objects 0x2110: Encoder 1 configuration and 0x2112: Encoder 2 configuration, subindex 28 Sampling frequency.

The sampling frequency can be set to a value between 16 or 32 kHz.

> Note:
> This is only available since firmware 5.0.7 In older versions, the sampling frequency is 16 kHz.

Limitations

If the encoder can't be sampled at the desired frequency, the error EncSmplT will be raised.

Some encoders only allow a particular frequency:

- Circulo internal encoders with multiturn support only 16 kHz.

In all other cases, the Circulo internal encoders will always be sampled at 32 kHz.

Parameters

Object ID	Name	Description
0x2110	Encoder 1 configuration	Subindex 28 selects the sampling frequency for encoder 1
0x2112	Encoder 2 configuration	Subindex 28 selects the sampling frequency for encoder 2

Commutation with Hall and ABI combined

Overview

An ABI encoder and a Hall sensor can be combined to function as a single encoder for commutation. This combines the benefits of a high resolution angle from the ABI encoder, with the absolute electrical position provided by the Hall sensor.

Requirements

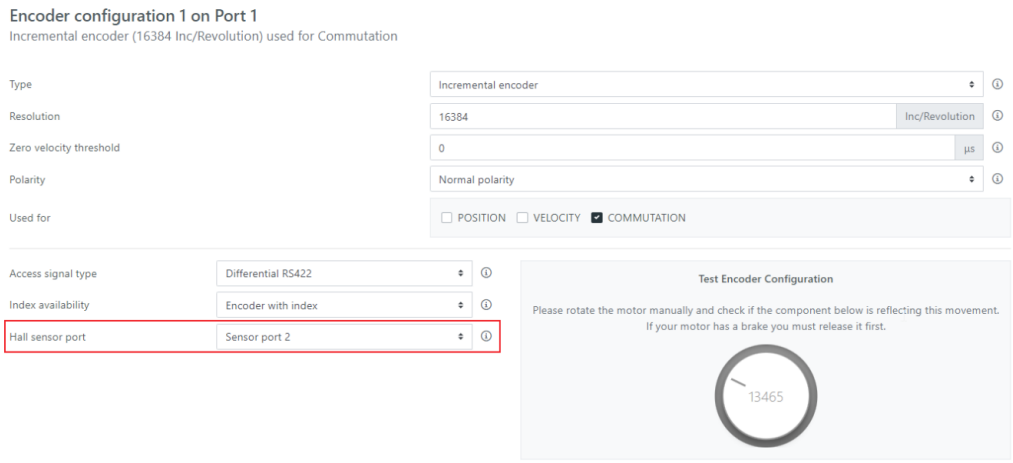
- The Drive needs a Hall sensor and an ABI encoder connected at two different connectors, both must be properly configured.
- The motor phases and the Hall pins have to be connected according to the Servo drive's documentation.
- The ABI commutation offset has to be found before this feature starts working.

Usage

To configure the HALL+ABI feature, the incremental encoder needs to be configured for at least "commutation" then specify the encoder port that the HALL sensor is connected to in the object "Encoder1 configuration Hall sensor port "0x2110:23"

The drive will automatically use it on startup to avoid the index detection routine. There is no need for additional configuration of the hall sensor on encoder port 2.

Configuring the Hall additionally on port 2 will lead to a HwRsrcEr.



Detailed description

Startup

When the feature is started, the Hall sensor is sampled and it is checked if it's in a valid state. If an invalid hall state [0, 0, 0] or [1, 1, 1] is found, an error 'HallSeq' is raised, and the connections should be checked.

Error Detection

Cable errors can lead to problems: if only one Hall sensor is disconnected or damaged, or if the phase signals are reversed, the problem will not be detected. Running the motor with Hall set as the only Commutation sensor can be done to ensure the integrity and proper connection of the Hall signals before integrating the ABI encoder.

Runtime

If no error is raised, the initial electrical angle is estimated based on the Hall sensors, and the Hall sensors are not sampled again. As soon as the incremental encoder passes the index, the service realigns the electrical angle to the formerly saved precise value.

Input counter

Overview

This feature allows counting pulses in a particular pin of an encoder connector (input counter pin).

It counts the transitions on the input counter pin at a maximum frequency of 16 kHz. The value of the counter is written to object 0x2040.

The input counter will be enabled by default in case the configuration allows it (for limitations see below).

Either a single-ended TTL line or a differential RS422 line can be used for the input counter.

Drive	Conditions for using a Single-ended (TTL) input counter	Conditions for using a differential (RS422) input counter
SOMANET Node	Encoder connector 2 doesn't have any encoder configured. Pulses are counted at pin 7.	Encoder connector 2 has an SSI encoder configured. Pulses are counted at pin 7 (-) and 8 (+).

Use case: Counting revolutions for simple speed estimation

In applications that run only in one direction (e.g.: a fan), it may be sufficient to count the pulses for calculating the speed.

Limitations

- At the moment, this feature can only be used if less than 2 encoders are configured or if one of the encoders is an SSI.
- If only one SSI encoder is connected (the other port being empty), the feature will use the empty port.
- The feature can be used at a maximum frequency of 16 kHz.

Parameters

Object ID	Name	Description
0x2040	Input counter	The value of the counter will be written to this object

Protection

Overview

This feature allows users to define limits for the following values:

- Undervoltage
- Overvoltage
- Overcurrent

Once these limits are violated, an error is triggered and the drive is stopped.

Also, a limit for Overtemperature triggers the error but this value is not customizable.

The temperature of the core board and the inverter are also measured and the values are available through objects 0x2030 (Core temperature) and 0x2031 (Drive temperature). The unit is given in m°C.

When the measured temperature approaches 95% of the permissible value, a warning is triggered in the Error Report Object. When the measured temperature reaches the limit, an error is triggered and the drive is stopped.

Usage

To set the value for user-defined protection limits, enter the value in the corresponding subitems of object 0x2006 Protection

- Undervoltage: 0x2006:1, Unit given in V
- Overvoltage: 0x2006:2, Unit given in V
- Overcurrent: 0x2006:3, Unit given in mA

Parameters

Object ID	Name	Description
0x2006	Protection	Limits for Undervoltage, Overvoltage and Overcurrent
0x2030	Core temperature	The temperature measured on the Core board
0x2031	Drive temperature	The temperature measured near the inverter

Errors and Warnings

To facilitate error diagnostics, SOMANET features several mechanisms:

Error codes thrown by the firmware SOMANET throws firmware errors as specified in CiA 402

Error report object Each occurring error is described in the Error Report Object

Remedies for protection errors How to resolve issues with over-current, over-voltage, under-voltage and over-temperature

Error codes thrown by the firmware

Object 0x603F throws error codes as standardized in CiA 402:

Object ID	Name
0x0000	NO_ERROR
0x2220	DEVICE_INTERNAL_CONTINUOUS_OVER_CURRENT
0x2250	DEVICE_INTERNAL_SHORT_CIRCUIT
0x2351	I2T_THERMAL_STATE_LOAD_LEVEL_WARNING
0x3130	PHASE_FAILURE
0x3131	PHASE_FAILURE_L1
0x3132	PHASE_FAILURE_L2
0x3133	PHASE_FAILURE_L3
0x3210	DC_LINK_OVER_VOLTAGE
0x3220	DC_LINK_UNDER_VOLTAGE
0x3331	FIELD_CIRCUIT_INTERRUPTED
0x4210	EXCESS_TEMPERATURE_DEVICE
0x4310	EXCESS_TEMPERATURE_DRIVE
0x5300	OPERATING_UNIT
0x6010	SOFTWARE_RESET_WATCHDOG
0x6320	PARAMETER_ERROR
0x7121	MOTOR_BLOCKED
0x7300	SENSOR

Object ID	Name
0x7500	COMMUNICATION

Note:

Whenever a firmware error occurs, there is a dedicated entry in the Error Report Object which will specify what triggered the error.

Object ID	Name	Description
0x603F	Error Code	Provides the error code of the last error which occurred in the drive device

Error Report Object

Provides further information about the error or warning reported by the Error code object 0x603F.

No fault

Error code	Description	Explanation	Error code from object 0x603F	Remedy
NoFault	No fault	Default value of the object, which means no fault is to be reported.		This error code should indicate that there's no error reported, thus the drive is not in fault state. If the drive is shown in fault state and this error code is reported, possibly

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				this is caused by a fault in master/slave communication for example in the VM version OBLAC Drives. Rebooting the VM would solve the problem in this case. Else please contact technical support of Synapticon.

Protection faults of user-defined parameters

Errors triggered by user-defined thresholds (for more details see object 0x2006: Protection)

Error code	Description	Explanation	Error code from object 0x603F	Remedy
PuOcA	Protection User-defined Over-current Phase A	The current sensor at drive phase A/B/C sensed a current exceeding the	0x2220	As the threshold is user-defined, please first check the object 0x2006:3 Overcurrent setpoint to make sure that the threshold is set

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		user-defined Over-current threshold (0x2006:3 Protection Over-current setpoint).If the user-defined threshold is higher than the hardware protection threshold, as long as the hardware protection is available, this software protection can be considered unfunctional.		appropriately.Regarding an over-current, please refer to the remedy for over-current.
PuOcB	Protection User-defined Over-current Phase B		0x2220	
PuOcC	Protection User-defined Over-current Phase C		0x2220	
PuOt	Protection User-defined Over-temperature		0x4210, 0x4310	
PuUv	Protection User-defined	Indicates that	0x3220	As the threshold is

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	Under-voltage	the DC bus voltage is measured below the user-defined threshold (0x2006:1 Protection Under-voltage setpoint) when the drive is in a CiA 402 state that requires high-level power to be applied. These states are Switched on, Operation enabled, Quick stop active and Fault reaction active).		user-defined, please first check the object 0x2006:1 Under-voltage setpoint.Then please refer to the remedy for under-voltage.
PuOv	Protection User-defined	Indicates that the DC bus	0x3210	As the threshold is user-defined, please

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	Over-voltage	voltage is measured exceeding the user-defined threshold (0x2006:2 Protection Over-voltage setpoint).		first check the object 0x2006:2 Overvoltage setpoint. Then please refer to the remedy for over-voltage.

Hardware Protection Faults

Errors triggered by hardware built-in protection logic.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
PhUdef	Protection Hardware Undefined	Indicates a malfunction of the protection circuit.	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				investigation.
PhFault	Protection Hardware Fault	A generic hardware fault was detected. In Circulo drives this can be caused by any of the following: phase over-current, under-voltage, over-voltage, over-temperature, deadtime error, shoot-through or watchdog timer.	0x5300	Please refer to the Protection hardware remedies for over-current, under-voltage, over-voltage, over-temperature, deadtime phase and watchdog.
PhWtdg	Protection Hardware Watchdog	The watchdog is a service, by checking periodically with a timer, guaranteeing that the drive control logic is working in the proper fre-	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		quency. This error indicates that the watch-dog tick is not received on time by the hard-ware protection, thus the drive is not working properly.		investigation.
PhDeadTA	Protection Hardware Deadtime Phase A	Deadtime is a common pro-tection strategy in PWM control for avoiding bridge shoot-through, a short-circuit accident caused by tran-sistors on both sides being open at the same time. This error indicates that the effect-ive deadtime is	0x2250	Please note down the information regarding the occurence of the error and reach Syn-apticon tech-nical support for further investigation.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		detected below the man-ufacturer-defined threshold.		
PhDeadTB	Protection Hardware Deadtime Phase B		0x2250	
PhDeadTC	Protection Hardware Deadtime Phase C		0x2250	
PhDeadTD	Protection Hardware Deadtime Phase D		0x2250	
PhOvUvOt	Protection Hardware Over-voltage / Under-voltage / Over-tem-perature	The drive tem-perature which measured close to power MOSFETs and the DC bus voltage are checked by the same hardware protection cir-cuit against manufacturer-defined	0x3331	Please refer to the remedy for over-voltage, under-voltage and over-tem-perature.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		thresholds. This error indicates that one or more of the three errors is/are triggered.		
PhOc	Protection Hardware Over-current	Indicate that the phase current or the DC bus current is measured exceeding the manufacturer-defined threshold (referring to the hardware document, link), which triggers the hardware protection circuit.	0x2220	Please refer to the remedy for over-current.

Safety Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SfeDiIvd	Safety Digital Input Invalid	Safety module reports a discrepancy between the two STO-SBC input channels. The error will be shown with a 100 ms interval.	0x5300	Check and fix the safety inputs wiring accordingly with the assistance of tools for example with multimeter measurement. Compare the measured inputs with 0x2611 (Safety digital input diagnostics) and locate the signal leading to the fault. Note that You can only reset the safety fault if you are in the SAFETY ACTIVATED state, meaning both STO-SBC1 (0x2611:1) and STO-SBC2

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				(0x2611:2) inputs shall be in the Low state when you are trying to clear the fault. Please refer to the documentation about the safety functions and the STO/SBC inputs wiring of your hardware and check the use of the safety functions in our documentation.
SfeFault	Safety Fault	A safety-based hardware fault detected.	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further investigation. If this fault is permanent and can't be reset this

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				is a fault caused by broken internal hardware

BiSS Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BisErBit	BiSS Error Bit active.	One of the bits in the BiSS frame is used to signal error. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the error bit being active.	0x7303, 0x7304	Please check the 2 cases often seen:1. Mis-configuration of Status bits active value. The status bits (error bit and warning bit) are by default read as active low, i.e. the error bit being 0 indicates the error being active. Please check the documentation of

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BisWnBit	BiSS Warning Bit active.	One of the bits in the BiSS frame is used to signal warning. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the warning bit being active.	0x7303, 0x7304	the encoder and correctly configure the reading.2. The encoder is indeed rasing an error/ warning. Please check the documentation of the encoder, consult the encoder manufacturer for further investigation.
BisAcBit	BiSS Acknowledge Bit not received.	The BiSS slave is not acknowledging the request to communicate sent by the drive. This usually happens when there is some	0x7303, 0x7304	Please firstly make sure that the encoder is connected to the port which is being configured. Then check the wiring accoding to the

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		issue with the power or communication link, e.g. one of the wires is broken or not well connected.		hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
BisSrtBt	BiSS Start Bit not received.	The BiSS slave acknowledged the request to communicate, but it didn't sent the start bit to start the communication. This usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	0x7303, 0x7304	
BisSloLo	BiSS data (SLO) is per-	The SLO line	0x7303, 0x7304	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	manently Low.	should be set to high by the BiSS slave when it is ready to establish communication. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.		
BisFrame	BiSS Frame configuration error.	The sensor configuration causes an issue related to the BiSS frame. This error is currently raised only if the BiSS frame size is higher than 64 bits.	0x7303, 0x7304	Please check the sensor configuration according to the documentation of the encoder manufacturer. The most relevant objects to check:1. Resolution, unit Inc/

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				Revolution2. Multiturn resolution, unit number of bits3. Number of filling bits, unit number of bits
BisCrc	BiSS CRC error.	The BiSS frame contains a CRC6 of some of the data that is being sent. This error is raised when the CRC sent by the BiSS slave is different than the CRC computed by the drive. This usually means that the drive can't read the frame properly.	0x7303, 0x7304	Regarding a CRC error, there are several causes often seen:1. Misconfiguration of the sensor reading leading to the bits being read wrong. In this case the fault cannot be reset. Please check the sensor configuration according to the documentation of the encoder

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				manufacturer. Common parameters wrongly configured are the singleturn resolution and multiturn resolution.2. Misconfiguration of the object CRC polynom, thus the CRC is calculated wrong. In this case the fault cannot be reset. Please check the polynom in the encoder documentation. The value by default: 48 represents x^0 + x^1 + x^6, which usually doesn't need to be modified.3. Inconsistency in

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
BisRegEr	BiSS register communication error	Some BiSS encoders require the firmware to read/write some of their registers to be initialized	0x7303, 0x7304	Try to reconfigure the encoder that raised this error. To do so, there are two possibilities:- Set its

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		properly. This error is triggered when any of these BiSS register transaction fail.		function to Disabled and then back to the function it had.- Remove the encoder from the sensor port list (0x2100) and add it again.

SSI Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SsiFrame	SSI Frame configuration error.	The sensor configuration causes an issue related to the SSI frame. This error is currently raised only if the SSI frame size is higher than 256 bits.	0x7303, 0x7304	Please check the sensor configuration, especially the object Frame size according to the documentation of the encoder manufacturer.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SsiDtaLo	SSI Data is permanently Low.	The data line should be set to high by the SSI slave when it is ready to establish communication. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	0x7303, 0x7304	Please firstly make sure that the encoder is connected to the port which is being configured. Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
SsiDtaHi	SSI Data is permanently High.	The data line should be set to low by the SSI	0x7303, 0x7304	Please firstly make sure that the encoder is

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		slave during the monoflop time, which happens right after clocking the complete SSI frame. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.		connected to the port which is being configured. Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
SsiChksm	SSI Checksum error.	Some SSI slaves include a ckecksum of the data in the SSI frame. Currently our drives can	0x7303, 0x7304	Regarding a checksum error, there are several causes often

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		be configured to read a parity bit from compatible SSI encoders. This error means the parity bit computed by the firmware is different than the one sent by the SSI slave.		seen:1. Mis-configuration of the sensor lead-ing to the bits being read wrong or the checksum being calculated wrong. In this case the fault cannot be reset. Please check the sensor con-figuration accord-ing to the documentation of the encoder manufacturer.2. Inconsistency in signal transfer. In this case the fault can some-times be reset, but would occur under certain cir-cumstances. Please check the

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				eration for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.

QEI Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
QeiLsTck	QEI Lost Ticks.	The number of ticks counted in two successive index pulses is different from the sensor resolution configured by user.	0x7303, 0x7304	There are several causes often seen: 1. The configured resolution is wrong. Please check the Resolution configured according to the encoder datasheet, and please pay attention to the conversion between units (Resolution = Cycles per Revolution * 4). 2. There's a loss of pulses in encoder reading, which can be due to: 2a. The encoder

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				being improperly mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and consult the encoder manufacturer for advices. 2b. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influ-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				oldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transferred with acceptable noise. 2c. The pulses are read with an exceeding frequency. This case rarely happens, but can be possible if the encoder resolution and/or the spin speed being unusually high.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
QeiNoIdx	QEI No Index.	For an incremental encoder configured to be with 3 channels (usually known as ABI/ ABZ), the drive is expecting to receive an Index pulse which provides an index for the absolute singleturn position. This error indicates that the Index pulse couldn't be detected, while the drive moved the motor in order to detect it.	0x7303, 0x7304	Please first check whether your encoder is indeed with the Index pulse, i.e. is an ABI/ ABZ encoder. If it is an encoder without index, please choose 2-channel (no index) for Number of channels. Else, most probable causes are: 1. The encoder being improperly mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				consult the encoder manufacturer for advices. 2. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transferred with acceptable noise.
QeiOpnWr	QEI Open Wire.	Indicate that one of the differential A or B lines is disconnected. Only available on hardware that supports it.	0x7303, 0x7304	Please check the wiring with for example the diode mode of a multimeter, making sure that the wires are stably transferring the signals.

Hall Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
HallSeq	Hall Sequence error.	The hall sensor outputs	0x7303, 0x7304	Please check

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		have an invalid value (111 or 000), or the sequence between the current state and the last state is invalid.		the wiring and make sure that the signals are transferred with good consistency. If the HALL sensor is used together with an incremental encoder for commutation, it can be that the HALL sensor A/B/C pins are connected in a wrong order. Please verify the cabling, and verify the order by running offset detection with only the HALL sensor configured for commutation. If the order is correct, the result of the offset detection

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				should be around 0, and 0x2003:5 Motor phases inverted should show 0 (No).

A-Format Sensor Fault

Error code	Description	Explanation	Error code from object 0x603F	Remedy
AftFrame	A-Format Frame error.	The fixed data in the recieved A-Format frame (like sync code) is wrong. This usually means that the A-Format transmission speed is wrong.	0x7303, 0x7304	Please check the transmission speed configuration according to the encoder datasheet.
AftTmout	A-Format Timeout.	The A-Format encoder didn't start communication after sending a	0x7303, 0x7304	Please check the wiring and make sure that the signals are transferred with

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		request.		good consistency. Use a oscilloscope to monitor the signal transferred for further investigation.
AftCrc	A-Format CRC error.	The A-Format frame contains a CRC of some of the data that is being sent. This error is raised when the CRC sent by the encoder is different than the CRC computed by the drive. This usually means that the drive can't read the frame properly, which usually happens because of noise in the com-	0x7303, 0x7304	Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situ-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		munication link.		ation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transferred with acceptable noise.
AftBatt	A-Format BATT status.	The encoder is raising Battery Alarm. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Battery Alarm. The usual case is that the battery voltage is low, the battery is wrongly wired, or there's no battery connected.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
AftMTErr	A-Format MTERR status.	The encoder is raising MT Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the MT Error.
AftOvFlw	A-Format OvFlow status.	The encoder is raising Over Flow Alarm. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Over Flow Alarm.
AftOvSpd	A-Format OVSPD status.	The encoder is raising Over Speed Alarm. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Over Speed Alarm. The usual case is that the rotation speed exceeds the speed limit

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				of the encoder.
AftMemEr	A-Format MEMERR status.	The encoder is raising Memory Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Memory Error.
AftSTErr	A-Format STERR status.	The encoder is raising ST Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the ST Error.
AftPSErr	A-Format PSERR status.	The encoder is raising PS Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the PS Error.
AftBusyF	A-Format BUSY status.	The encoder is raising BUSY Flag. Check your	0x7303, 0x7304	Please check the encoder documentation for

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		A-Format encoder data-sheet for more information.		better under-standing the meaning of the BUSY Flag.
AftMemBs	A-Format MemBusy status.	The encoder is raising MEMBUSY Flag. Check your A-Format encoder data-sheet for more information.	0x7303, 0x7304	Please check the encoder doc-umentation for better under-standing the meaning of the MEMBUSY Flag.
AftOvTmp	A-Format OvTemp status.	The encoder is raising Tem-perature Warn-ing. Check your A-Format encoder data-sheet for more information.	0x7303, 0x7304	Please check the encoder doc-umentation for better under-standing the meaning of the Temperature Warning. In case of over-temperature please refer to the remedys for over-tem-perature.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
AftIncEr	A-Format IncErr status.	The encoder is raising Incremental Signal Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Incremental Signal Error.

Generic Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SnsrAngl	Sensor Angle error.	1. The electrical angle was not completing a whole revolution during offset detection. 2. The calculated commutation angle is out of bounds. A sing-	0x7300, 0x7303, 0x7304	Please firstly check whether the motor rotates during offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. Please check

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		leturn related configuration parameter (resolution, singleturn) with a wrong value normally causes this error. 3. The motor is not moving during the QEI index detection.		the encoder configuration, and make sure that while rotating the motor for one full turn, the encoder value is correctly responding to the mechanical position. You can manually rotate the motor for investigation. An alternative way is with OBLAC opening another page and ploting the position feedback of the encoder used for commutation while running offset detection.
SnsrCfg	Sensor Configuration error.	1. Some	0x6320, 0x7303,	Please check the configuration

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		encoder parameter was set with an invalid value in the Object dictionary. 2. An action that required a particular encoder function (position, velocity or commutation) was triggered, but there is no encoder configured for that function. 3. There was an issue when trying to initialize or configure an encoder.	0x7304	of the sensors, especially the parameters changed lately. Make sure that all functions required (position, velocity, commutation) are configured with the sensors.
HwRsrcEr	Hardware Resource Error.	The software configuration is invalid for the	0x7303, 0x7304	Please check the sensor configurations and

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		hardware. For example, a sensor configured is with a sensor type not supported by that port.		the digital IO settings, making sure that the setting is supported by your hardware according to the documentation.
IvldGpio	Invalid Gpio configuration.	The GPIO related configuration chosen is not available. This includes voltage level, direction of the GPIO, or GPIO trigger.	0x7303, 0x7304	Please check the configuration of the GPIO according to the documentation.
SnsrBatt	Sensor battery error	This error is raised when an encoder battery was expected, but couldn't be detected. This error is raised	0x7303, 0x7304	Please check the documentation to make sure the battery is connected properly and power cycle. In case there is

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		only by multiturn encoders that need a battery, like Circulo's Internal encoder 1 when it's configured as a multiturn encoder. The error is also raised when the multiturn count of such an encoder configured for multiturn is lost.		no battery or if the battery is not needed, select "0" or "No" to the "Multiturn resolution" parameter of this encoder.
SkpCycls	Serial encoder service is skipping cycles	The error SkpCycls is raised when a serial encoder service (REM16, BiSS, SSI, A-Format, MA302) is looping slower than	0x7303, 0x7304	Minimize the time that is required for reading the encoder. In a BiSS encoder the following approach can be used:

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		16 kHz, which means that the cycles are taking longer than 62.5 us. The warning SkpCycls is raised when an encoder that supports 32 kHz sampling (which are only BiSS or SSI) is looping at 16 kHz instead.		First the clock frequency should be increased, and then the encoder timeout should be reduced (determine the minimum value from the encoder datasheet). For an SSI encoder, only the clock frequency can be increased because the timeout is fixed. For other encoders, the speed is fixed. Please contact support if this issue occurs.
EncMtLos	Encoder multiturn was lost	This error is raised when the multiturn	0x7303, 0x7304	Please check the documentation to

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		count of a multiturn encoder that needs a battery is lost, like Circulo's Internal encoder 1 when it's configured as a multiturn encoder. This situation will happen when the battery of the encoder is low or removed and the drive is power-cycled.		make sure the battery is connected properly and power cycle. In case there is no battery or if the battery is not needed, select "0" or "No" to the "Multiturn resolution" parameter of this encoder.

Brake Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BrkNtRls	Brake release failed	Brake release	0x7121	Please firstly

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		procedure for pin brake failed, i.e. the configured displacement threshold (0x2004:8 Minimum displacement (pin brake)) is not reached with the configrued torque (0x2004:9 Maximum torque (pin brake)).		check whether the motor functions correct, for example by manually releasing the brake and running offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. While the motor functions correct, please further tune the two configuration parameters, by increasing the torque or decreasing the minimum displacement, until the brake can be released stably

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BkHiPull	Pull brake voltage too high	Pull brake voltage (0x2004:1 Pull voltage) higher than measured DC bus voltage (0x6079: DC link circuit voltage).	0x6320	Please measure the main power supply voltage supplied to the drive, and check whether the result is close to 0x6079: DC link circuit voltage. Pay attention that the actual voltage supplied to the drive would be slightly lower than the voltage output by the power supply, because of attenuation. Check the configuration of the pull/hold voltage, which should be lower than the DC Bus voltage.
BkHiHold	Hold brake voltage too high	Hold brake voltage (0x2004:2 Hold voltage) higher than measured DC bus voltage (0x6079: DC link circuit voltage).	0x6320	

Parameter Fault

Error code	Description	Explanation	Error code from object 0x603F	Remedy
IvldOfst	Invalid offset	The commutation offset in 0x2001: Commutation angle offset is not valid. This error comes directly from reading object 0x2009:1 Commutation offset: State.	0x6320	Please run offset detection before operations. Please pay attention that, the change in some fundamental configuration parameters (sensor configuration, motor configuration) can make the existing offset detection result invalid. Please redo offset detection in that case.
IvldOpmd	Invalid operational mode	Requested opmode is not supported. Set a valid value in 0x6060: Modes of operation.	0x6320	Please check our documentation for the available operation modes. Please check 0x6060: Modes of operation, and

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				make sure that the master controller is sending a correct command.
ZeroMxI	Max. current is zero or negative	User-defined limit on motor max current (0x6073: Max current) is 0.	0x6320	Please configure the 0x6073: Max current to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxT	Max. torque is zero or negative	User-defined limit on motor max current (0x6072: Max torque) is 0.	0x6320	Please configure the 0x6072: Max torque to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxMS	Max. motor speed is zero or negative	The value of 0x6080: Max motor speed is 0, so no motion is allowed.	0x6320	Pleaes configure 0x6080: Max motor speed to a valid value larger than 0, according

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				to your motor datasheet or your application need.
ZeroMxP	Max. power is zero or negative	User-defined limit on Max. power (0x200B) is 0.	0x6320	Pleaes configure 0x200B: Max power to a valid value larger than 0, according to your motor datasheet or your application need.
IvldPara	Invalid Parameter	The value of a parameter is invalid.		Please check the values changed lately according to the documentation.
SwLimOut	Software Limit Out.	0x607D: Software position limit is outside of encoder range limit.	0x6320	Please check object 0x607D: Software position limit and make sure that it is inside of 0x607B: Position range limit.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SiVelPfx	Invalid velocity scaling factor	The velocity scaling prefix defined by 0x60A9: SI unit velocity is outside of the supported range.	0x6320	Please check OBLAC Drives, which directly provides available options for the configuration of 0x60A9: SI unit velocity, and the documentation for user defined velocity units. Make sure that a valid data is write to the object.
SiVelTim	Invalid SI velocity time factor	The velocity scaling prefix defined by 0x60A9: SI unit velocity is outside of the supported range.	0x6320	Please check OBLAC Drives, which directly provides available options for the configuration of 0x60A9: SI unit velocity, and the documentation for user defined velocity units. Make

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				sure that a valid data is write to the object.
IvldFeed	Invalid Feed constant	One of the Feed parameters 0x6092 (Feed or Shaft revolutions) is set to 0.	0x6320	Please check the values of 0x6092:1 and 0x6092:2.

Motion Control Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
MxPwrLmt	Maximum power limit	Maximum power limit is active. Torque set point is limited.	0x5200	
FollwErr	Following error		0x8611	
MxTorLmt	Maximum torque limit	Maximum torque limit is active. Target or demand torque is lim-	0x5200	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		ited.		
MxMSLmt	Maximum motor speed limit	Max. motor speed limit is active. Target or demand velocity is limited. Also, it is active when Profile velocity in Profile postiion mode is limited.	0x5200	
SwPosLmt	Software position limit	Software position limit is active. Target position is limited or Actual position is limited in order to prevent runaway of a motor.	0x5300, 0x8612	
PosRgLmt	Encoder range limit	Encoder range limit is active. Target position is limited and can't wrap around range	0x8612	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		limits.		
IntgrLmt	IntLmAct	Integral limit is active. Integral part of position or velocity PID controller is limited.	0x5200	
PosLmAct	Positive Limit Switch Active	Motor is at the position limit switch and can't move further.	0x5300	
NegLmAct	Negative Limit Switch Active	Motor is at the negative limit switch and can't move further.	0x5300	

Communication Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
CyclicHb	Cyclic heartbeat	A communication	0x7500	If the inactivity of

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		keep-alive heart-beat was dropped. Check that the PDO messages are being sent by the master cyc-lically, and within 100 ms of each other.		communication is known/ inten-ded, the occur-rence of the error is expec-ted. Do [Fault reset] to clear the fault after the communication is reestablished. Otherwise, the occurrence of the error indic-ates that there exists unknown issues in the communication chain. Possible causes include: - communication delay in master program, check the running status of the master machine. - disturbance in communication signal transfer,

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				make sure that a communication cable with shielding is used or try with the cable provided by Synapticon, check the stability of cabling.
OsCmdCol	OS command collision	A new OS command was sent, but the OS command response of a previous OS command was not read after it completed.	0x7500	In order to send another OS command, please read the response of the previous OS command in the object 0x1023:3. Whenever the first byte becomes 0, 1, 2 or 3, a new OS command can be sent.

I2t Protection Errors

Error code	Description	Explanation	Error code from object 0x603F	Remedy
I2tActv	I2t protection is active	I2t protection is active, which limits your maximum torque to the nominal rated values of the motor. See 0x200A: I2t for configuration parameters.	0x2350, 0x2351	Please check 0x200A: I2t and see whether it's configured as expected. Regarding the I2t being triggered, that indicates that the drive is protecting the motor from potential overheating. Please read our documentation of I2t for more information.

Homing errors

Error code	Description	Explanation	Error code from object 0x603F	Remedy
HmInvMth	Invalid homing method	The value of 0x6098: Homing method is invalid or not supported.	0x6320	Please firstly check object 0x6098: Homing method and see

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				whether the object is written as expected. Please refer to our documentation Homing modes, and check what homing methods are supported.
HmIvSwSt	Invalid Switch State during homing	The state of the homing switch at the end of the homing routine was unexpected. This can happen if the homing switch deactivates before the axis moves back into the valid range of motion.	0x5300	Ensure that the homing switches remain active after the mechanics hit them. Increasing the homing acceleration (0x609A) may help prevent overshoot after the switch is activated.

Internal Errors

Error code	Description	Explanation	Error code from object 0x603F	Remedy
Intern01	Internal fault 1	Internal Error Code. Please contact support.	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further investigation.
Intern02	Internal fault 2	Internal Error Code. Please contact support.	0x5300	
Intern03	Internal fault 3	Internal Error Code. Please contact support.	0x5300	
Intern04	Internal fault 4	Internal Error Code. Drive service is skipping cycles and there is no downstream communication.	0x7500	Please contact support.
Intern05	Internal fault 5	Internal Error Code. Motion service is skipping cycles and there is no downstream communication.	0x7500	Please contact support.

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

Filter cutoff frequency bounds

Error code	Description	Explanation	Error code from object 0x603F	Remedy
VeFiFcLo	Velocity filter cut-off frequency too low	0x2021:2 Velocity feedback filter Cutoff frequency is lower than the minimum limit.	0x6320	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
VeFiFcHi	Velocity filter cut-off frequency too high	0x2021:2 Velocity feedback filter Cutoff frequency is higher than the maximum limit.	0x6320	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				2000 Hz. Please check the documentation of the object for latest information.
PoFiFcLo	Position filter cut-off frequency too low	0x2022:2 Position feedback filter Cutoff frequency is lower than the minimum limit.	0x6320	Please check 0x2022:2 Position feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
PoFiFcHi	Position filter cut-off frequency too high	0x2022:2 Position feedback filter Cutoff frequency is higher than the	0x6320	Please check 0x2022:2 Position feedback filter: Cutoff

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		maximum limit.		frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
VffFcLo	Velocity feed forward filter cut-off frequency too low	0x2015:2 Velocity feed forward Cutoff frequency is lower than the minimum limit.	0x6320	Please check 0x2015:2 Velocity feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
VffFcHi	Velocity feed forward filter cut-off frequency too high	0x2015:2 Velocity feed forward Cutoff frequency is higher than the maximum limit.	0x6320	Please check 0x2015:2 Velocity feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
NhFiFcHi	Notch filter center frequency too high	0x2023:2 Notch filter Center frequency is higher than the maximum limit.	0x6320	Please check 0x2023 Notch filter, and make sure that the center frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest inform-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				ation.
NhFiPara	Invalid Notch filter parameters	Invalid configuration was detected for the notch filter.	0x6320	Please check 0x2023 Notch filter, and make sure that the rejection band is set smaller than the center frequency.

General errors

Error code	Description	Explanation	Error code from object 0x603F	Remedy
InvConf	Invalid Configuration	The operation to save the config.csv file did not properly complete, and the config.csv may be corrupt. The config.csv file will not be loaded.	0x6320	This most likely happened as a result of power cycling the drive directly after issuing a command to save the configuration. A file named "lock"

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				symbolizes that the save operation wasn't completed. If you reconfigure your drive and successfully save the configuration, operation will be restored. You may also upload a known good config.csv file, delete the "lock" file, issue a "restore configuration" command.
SkpCycls	Motion or drive service is skipping cycles	The warning SkpCycls is raised when a motion or drive service is looping slower than 4 kHz, which	0xF002	Try to disable some features (e.g. notch filter or position feedback filters) and see if the problem is fixed.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		means that the cycles are taking longer than 250 us.		
CtrlCyEx	Control cycle time is exceeded	This warning is raised when a Somanet control loop is executing slower than 4 kHz, which means that the cycles are taking longer than 250 us.	0xF002	Try to disable some features (e.g. notch filter or position feedback filters) and see if the problem is fixed.

Temperature Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
TmpCore or PhOtCore		This fault is triggered when the temperature of the core	0x4210	Disable the drive until the temperature is reduced or use external cool-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		becomes too high and crosses the error threshold (> 100 °C). This happens usually because of sustained load on the processor, high operating frequency and even high ambient temperature. High core temperature can lead to equipment damage. It can also result in irregular clock timing.		ing to bring down the overall temperature.
TmpDrive or PhOtDrve		This fault is triggered when the temperature of the drive becomes too high and crosses the error	0x4310	Disable the drive until the temperature is reduced or use external cooling to bring down the over-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		threshold (> 100 °C). This happens usually because of sustained load on the processor, current leakage from the onboard componenets, high operating frequency and even ambient temperature. High drive temperature can lead to equipment damage.		all temperature.

External Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
ExAnSnsr	External Scaled Analog Sensor	This fault is triggered when	0x7300	Please make sure that the

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	Output exceeded one of the threshold values.	the scaled external analog sensor output exceeds the upper level threshold or becomes lower than the lower level threshold. 0x2038:11 External scaled measurement Upper error threshold 0x2038:12 External scaled measurement Lower error threshold		upper and lower level threshold values are suitable for the sensor you have connected and that the values have been calculated according to the instructions given in the documentation for scaled measurements.

Remedies for protection errors

Remedy for over-current

There are various possible causes for an over-current error. Here are several potential ones which would be helpful to check:

- the controller output results in an exceeding current generated in the phase. Read the 0x6077 Torque actual value and the 0x6071 Target torque during operation, the scope of OBLAC can be used for this purpose. If an exceeding value can be seen in the torque value while the error happens, it often indicates that:

    - the PID controller (of the loop where the error is seen) is tuned too sharp to be stable. In this case, tune the controller softer following our tuning instructions. The controller can be easily unstable if the physical stability of the setup is bad, for example when the setup is not well fixed, or the load is with low rigidity.
    - the command sent to the drive is too sharp. Make sure that your command is sent with an appropriate acceleration / deceleration value. If a position step profile with relatively large amplitude or a quick drag of the slider in Playground-Position is given, this can be the case.
    - if the velocity value doesn't follow the torque value well (the torque doesn't lead to active acceleration) there may be a problem in the commutation angle calculation. Check the feedback value of the sensor and perform the offset detection again.
- an inappropriate power supply is used:
    - a voltage too high or too low would lead to malfunction of components. Check the voltage output of your power supply, if necessary with a measurement tool.
    - if the power output rating of the supply is not enough for the application, the voltage would drop while current is high, leading to the issue declared above.
- a short in the circuit can easily trigger over-current.
- a damaged hardware. In this case, there are often visible symptoms such as sparks or smoke.

Remedy for over-voltage

Possible causes for over-voltages include:

- the voltage output by the power supply is indeed exceeding the threshold. Adjust the voltage output accordingly.
- the voltage jump during motion leads to the error:

  When a load is causing the motor to turn in the opposite direction of the commanded motion, the motor generates power instead of consuming power. This leads to a voltage jump if there's no braking chopper in the circuit.

In this case use a power supply with integrated chopper or add a chopper device in the circuit for protection (e.g. Synapticon braking chopper).

Please also refer to our guidelines about regenerative energy.

Remedy for under-voltage

Possible causes for under-voltage include:

- the voltage supplied to the drive is continuously below the defined threshold.

> Note:
>
> It is normal to have a small (<1V) drop in voltage in the circuit due to resistance.

The power supply should be set to output a voltage slightly higher than the threshold or the threshold should be set lower in this regard. For better investigation, measure the voltage with a multimeter:

1. 0x6079 The DC link circuit voltage
2. The voltage between the main power supply pins on the drive
3. The voltage output by the power supply

The three results should be close. Otherwise, if there are large difference between the first two results, the hardware may be damaged. If the first two results are close, and the third is different, there may be a big voltage drop in the supply circuit due to factors such as cable resistance.

- If the error occurs during motion, the voltage output of the power supply drops when the maximum output current is reached. Check the specification of the power supply and make sure that a supply with adequate power rating is used.

Remedy for over-temperature

When the temperature of the drive is too high, it is usually a sign of inadequate cooling. First, please check the values in 0x2030 Core temperature and 0x2031 Drive temperature and note down the values. If the values seem much higher than a reasonable value, please contact Synapticon technical support. Please refer to our thermal installation guide for further information about improving the cooling.

Logging

Overview

Logging provides a list of events that happened within the firmware so that when an error or warning occurs, the user can trace back and see what might have triggered the error.

Operational activity logs of SOMANET systems can be generated and stored for bug-fixing, troubleshooting and diagnosis of systems.

The logs can be accessed via FoE:

- the current logs are saved to log_curr.log **log_curr.log** which has a capacity of 9 kB.
- When its memory is filled up, the logs are transferred to **log_prev.log** so that the latest 9 kB of logging history can always be retrieved.
- By concatenation of 'log_curr.log' and 'log_prev.log', between 9 kB and 18 kB of logging history can be retrieved.

Types

Logs are tagged with a "type" to facilitate quickly searching and finding important events within the log. These types are as follows:

DEBG	denotes a debugging log entry
INFO	this is the default log level. It contains information such as state changes
WARN	warnings, no fault is triggered
ERR	error, drive fault expected

All standard and SOMANET specific errors are listed in the section Errors and Warnings

Format

Each line in the log has the format:

<time TYPE|><message>

If a message exceeds 105 characters, the message is truncated and a ~character is placed in the 105th position.

The time initially starts at *00:00:00.000* and is thus the time since last boot.

Motion Control

To make you motion system perform in an optimal way, it is necessary to understand the basic concepts behind motion via EtherCAT, more specifically the CiA 402 Drive Profile and the SOMANET specific extensions:

Control Loops

Current and Torque Control Loop Details and working principle about the current control loop

Velocity Control Loop Details about the velocity loop structure and the applied filters

Position Control Loop Details about the cascaded position loop structure and the applied filters

Dual Loop Cascaded Position Control For precise position control, it is possible to use two encoders for each cascade

Modes of operation

Cyclic modes

Cyclic synchronous torque control (CST) In torque control mode, the master sends desired torques to the drive

Cyclic synchronous velocity control (CSV) In velocity control modes, the master sends desired velocities to the drive

Cyclic synchronous position control (CSP) In position control modes, the master sends desired positions to the drive

Profile modes

Profile position mode In profile position mode, the trajectory to a target point is generated on the drive

Profile velocity mode In profile velocity mode, the motor will rotate at a target velocity

Profile torque mode In profile torque mode, the drive sends a target torque to the motor

Homing modes

Diagnostics opmode Used for triggering diagnostic OS commands

Switching the modes Describes how the modes can be selected

Control Loop Tuning Each motor needs to be tuned before it is operational. Synapticon provides an easy-to-use Auto-Tuning feature but all drives can also be tuned manually

Extended control functionalities These features can be used to enhance performance

Cogging compensation Removes ripples caused by magnetic interaction between motor rotor and stator

Field weakening The velocity range of electric motors can be extended by weakening the magnetic field of the rotor linearly over the speed range

Anti-windup control An Anti-Windup algorithm is used to prevent the Integrator part to accumulate beyond acceptable boundaries

Gain scheduling Applies controller gains, which will change dynamically regarding the current speed

Motor overload protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating

Velocity feed forward To increase the velocity of the robot's reaction, the target position signal can bypass the position controller

Torque offset A desired offset torque can be added to the reference torque

Velocity offset An additional velocity offset can be used to improve the positioning performance

Filtering Digital filters can be used to remove unwanted frequencies such as noise or nonlinear effects

Position and Velocity Feedback Low-pass filters

Torque loop input shaping filter

Command smoothing and Interpolation Successive setpoints can be smoothed, so that low command resolution and jitter have a reduced effect on the control

Control supervision Allows to supervise the work of the control loops

Distributed clocks (DC clocks) Allows to synchronize the system time for enhanced performance

OS Command Offers commands that can't be executed by single SDO operations

Quick stop Stops the motor quickly in a controlled way

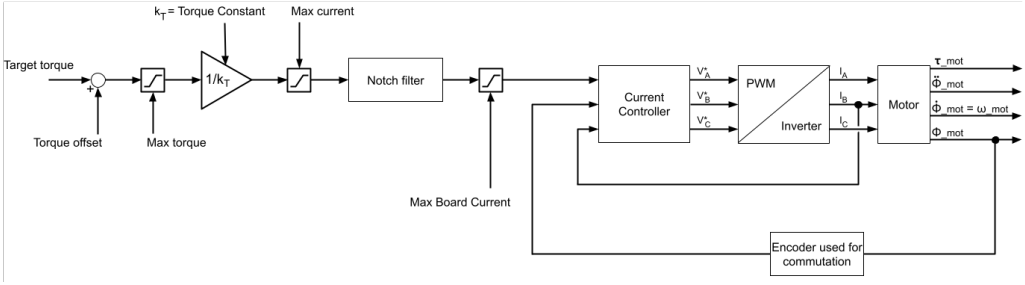
Control Loops

- Current and Torque Control Loop
- Velocity Control Loop
- Position Control Loop
- Dual Loop Cascaded Postion Control

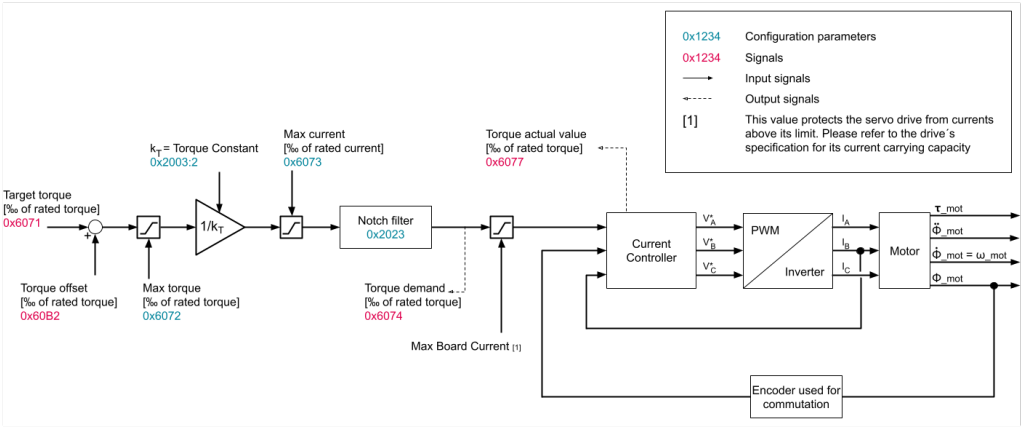
Current and Torque Control Loop

Torque control loop

General view



Detail view



> Note:
>
> The master commands a torque but internally it is computed as current. In most cases, as the position or velocity is the controlled value, the current measurement is sufficient. If a precise torque is required by the application, it should be verified with a torque sensor.

General Scope of the Torque Controller

The task of the torque controller is to make sure that the torque currently produced by the motor corresponds to the desired torque. This makes it one of the core algorithms of servo drives. It is used whenever a motor is driven.

In general, the RMS value of motor phase currents are proportional with the generated torque. The ratio between these two is known as **torque constant k_T** (as defined in subitem 0x2003:2).

The torque control loop is executed at a frequency of 16 kHz (as of firmware v4.2.0).

Reference Torque Generation

The reference torque can be updated in every execution step. In cyclic synchronous torque mode, the reference torque is transferred directly from the EtherCAT master to the torque controller via the PDO (object 0x6071). In other modes, a position or velocity controller is running on the

SOMANET Drive and acts as higher level control, so their outputs act as reference for the torque controller. The offset torque 0x60B2 is also part of the reference torque as described in Torque Offset.

Functional Description and Controller Structure

The following figure represents the structure of the torque controller. The implemented torque control algorithm is based on field oriented control (FOC).

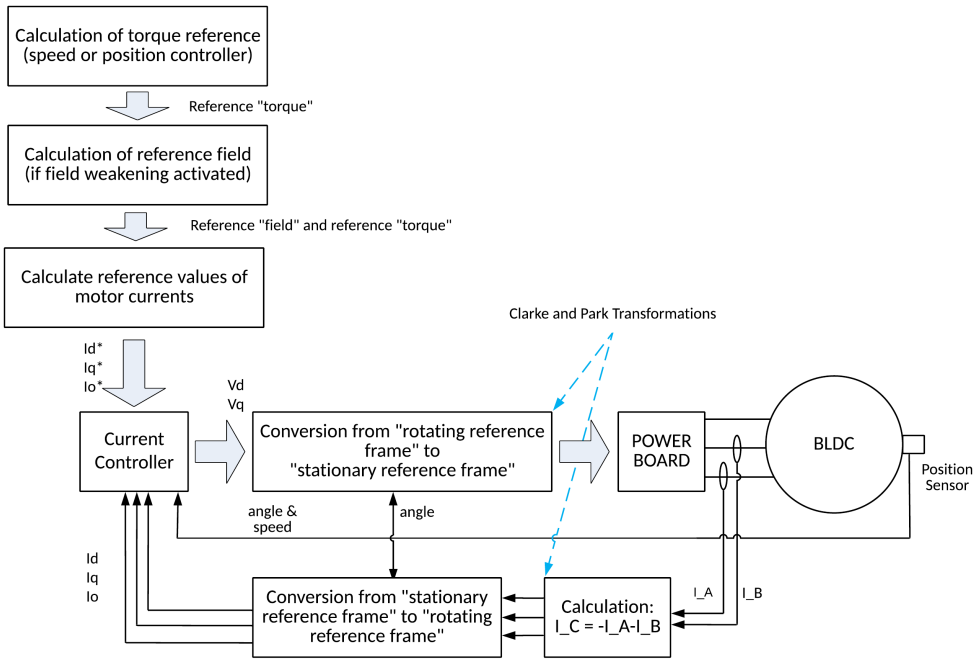


Figure 3: Torque loop structure

Tuning of the current controller

The controller can be fine-tuned by directly setting two parameters:

- Settling time (subitem 0x2010:10)
- Damping ratio (subitem 0x2010:11)

Settling time is entered in [µs], the default value 1000 µs.

Damping ratio is entered in [per mill], the default value is 2000 ‰.

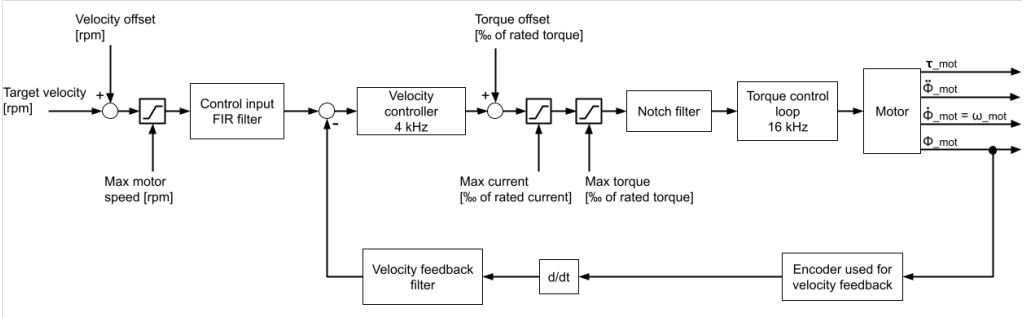
> Use case:
>
> For a damping ratio of 0.7, the entered value should be 700 [per mill].

Parameters

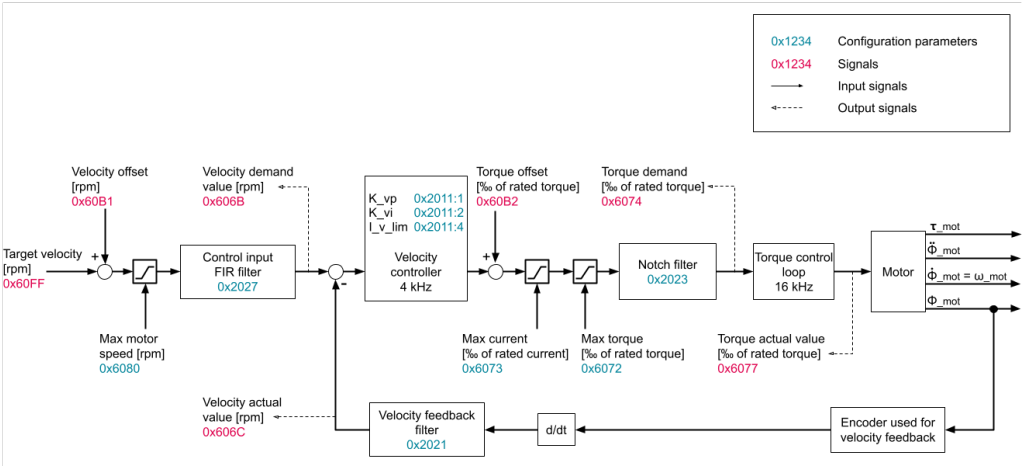
Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller

Velocity Control Loop

General view



Detailled view



The internal structure of the velocity controller is a PID feedback structure, along with anti-windup on the integrator and low-pass filter on the velocity feedback signal to deal with the sensor noise. **It is recommended to use this controller only as a PI controller (leave the D gain set to 0).**

On top of the desired velocity input, the velocity controller also accepts an additional velocity offset and additionaly, a torque offset can be added to the control loop to improve the control performance. This dynamically changeable torque value will be added to the torque value generated by the velocity PI controller.

Although the cascaded position control loop also includes a velocity feedback loop, the velocity controller described here and especially its controller gains are not used as part of any position

controller. This means the controller gains for the velocity controller described here (objects 0x2011:1 … 0x2011:4) and the velocity cascade being part of the position controller (objects 0x2012:5 … 0x2012:8) are set independently.

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2011	Velocity Controller	Values Kp, Ki, Kd and Controller Integral Limit for Velocity Controller

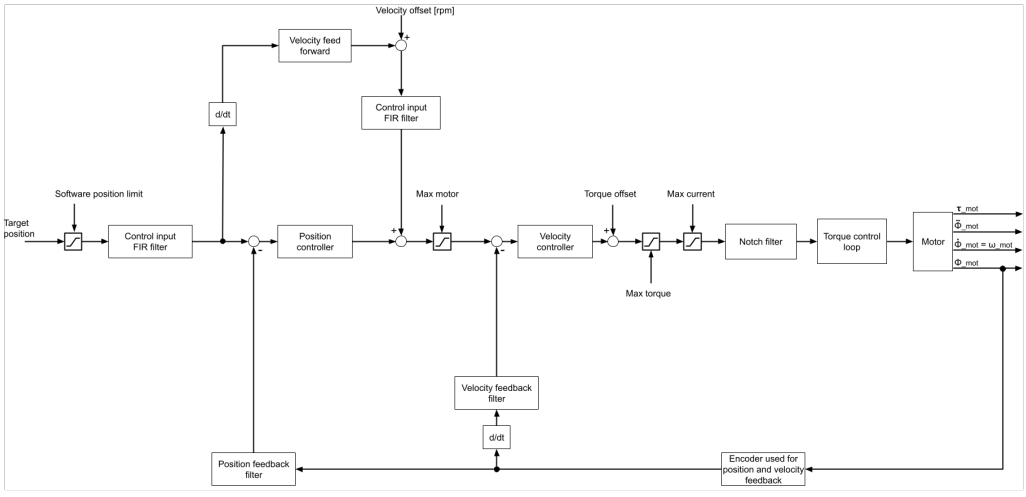
Position Control Loops

There are two implementations available for position control:

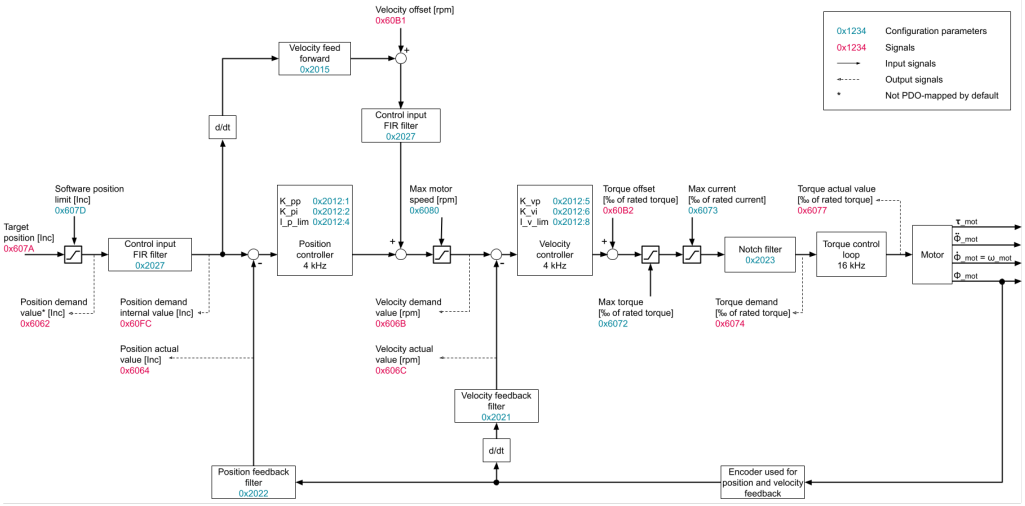
- Cascaded PID controller
- Simple PID controller

Cascaded PID Controller

General view



Detailled view



Note:

The specified units of the PDOs are valid only for the default unit configuration.

If user defined velocity (e.g. mRPM, rad/s) or position units are configured, they will be used instead. Additionally, in this particular configuration, the following objects: 0x606C Velocity actual value, 0x60B1 Velocity offset and 0x606B Velocity demand value are defined in the gearbox output reference frame, not in the motor frame.

See more about the velocity reference frame in the Migration Guide

Attention:

The velocity FIR filter is before velocity feed forward until firmware 5.0.3! Only in the soon-to-be-released firmware 5.0.4 it will be at the shown position.

The cascaded position controller consists of two PID feedback loops. The outer loop is used to compare the target position (current position setpoint) with the actual position value. From this difference, the PID controller computes a target velocity. The inner loop will compare this target velocity with the currently measured velocity and feed this difference to another PID controller that generates a target torque from this. This target torque is then controlled by the torque / current control loop described in the section on Torque Control. The overall control structure including the underlying torque control loop, the relevant update frequencies and the object numbers is shown in a simplified form in the figure above.

Although the two control loops offer a total of 8 controller parameters (P-gain, I-gain, D-gain and an integrator limit), it is not useful to use all of them at the same time. Generally, none of the D-gains should be used in the cascaded controller and the two I-gains should be used only alternatively, not together. This means that the control structure is simplified to a combination of one P controller and one PI controller: PI-P or P-PI.

Instead of the cascaded PID controller, OBLAC Drives allows the user to activate the simple PID position controller. This controller represents the simplest and most known form of position control. However, for productive use Synapticon recommends to use the cascaded controller. In the simple PID controller, one Proportional, Integral and Derivative feedback acts on the position itself and will directly generate the target torque / current.

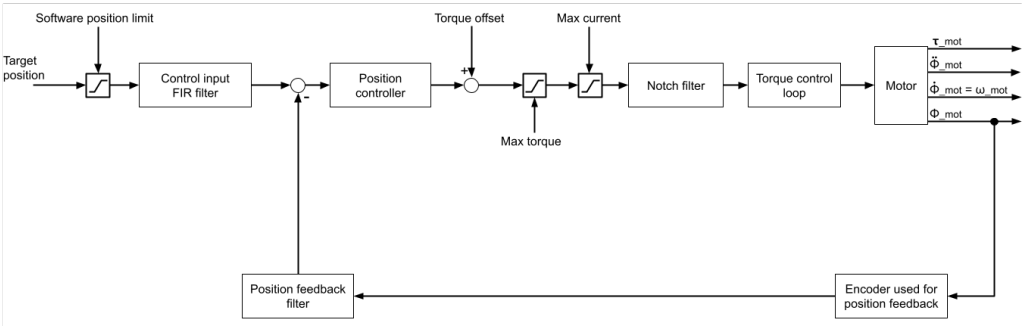
On top of the desired position input, both position control implementations also accept a desired 'offset torque', which is a dynamically changeable torque value that will be added to the torque value generated by the velocity PI controller. For further reference see Torque Offset.

Parameters

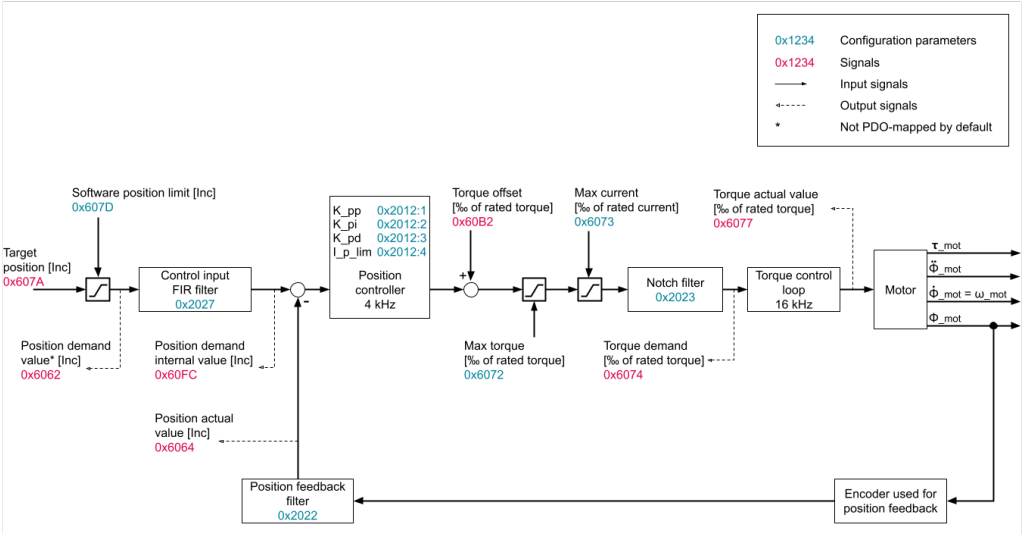
Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)

Simple PID Controller

General view



Detailled view



The cascaded PID controller is generally preferable over the simple PID controller. It is activated by default and apart from academic purposes there are few reasons to switch to the simple PID controller. However, the user can decide which position control structure is preferable for the particular application scenario.

Parameters

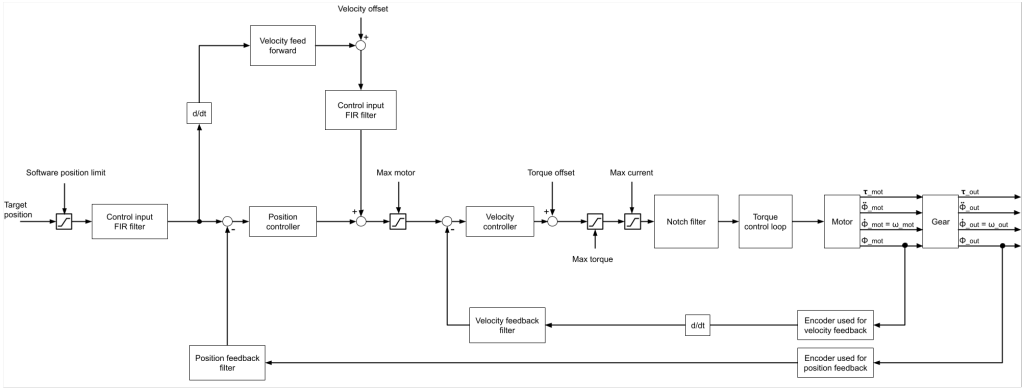
Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Simple PID Controller (Only subitems 1-4)

Dual Loop Cascaded Position Control

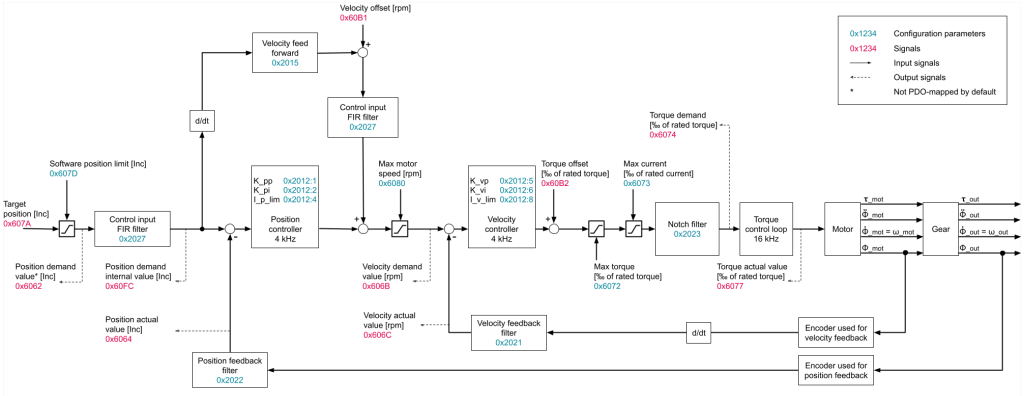
Overview

Using a gear can cause inaccuracy when only the shaft is controlled.
A second encoder can be attached to the output shaft after the gear for an optimized position control.
This section shows exemplary how the cascaded position controller can be configured for a dual encoder setup.

General view



Detailled view



Note:

The specified units of the PDOs are valid only for the default unit configuration.

If user defined velocity (e.g. mRPM, rad/s) or position units are configured, they will be used instead. Additionally, in this particular configuration, the following objects: 0x606C Velocity actual value, 0x60B1 Velocity offset and 0x606B Velocity demand value are defined in the gearbox output reference frame, not in the motor frame.

See more about the velocity reference frame in the Migration Guide

Attention:

The velocity FIR filter is before velocity feed forward until firmware 5.0.3! Only in the soon-to-be-released firmware 5.0.4 it will be at the shown position.

Usage

- Set the first encoder at the input shaft to *Commutation* and *Velocity*
- Set the second encoder at the output shaft to *Position*
- Ensure that the gear ratio is set correctly in object 0x6091.
- Set the encoder installed on the motor side (gear input shaft) to Commutation and Velocity functions.
- Set the encoder at the gear output shaft to Position function.
- More details on encoder configuration can be found here.
- Ensure that the gear ratio is set correctly in object 0x6091.

Parameters

Object ID	Name	Description
0x6091	Gear Ratio	Object for indicating the revolutions of motor and shaft
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)

Modes of Operation

SOMANET supports different modes of operation that either utilize a trajectory controller on the slave (e.g. profile modes) or get their trajectories from a master and cyclically decrease the difference between target and actual values (cyclic modes). Additionally, Homing modes are available to move the axis to a specific position.

Cyclic modes

Cyclic synchronous torque control (CST) In torque control mode, the master sends desired torques to the drive

Cyclic synchronous velocity control (CSV) In velocity control modes, the master sends desired velocities to the drive

Cyclic synchronous position control (CSP) In position control modes, the master sends desired positions to the drive

Profile modes

Profile position mode In profile position mode, the trajectory to a target point is generated on the drive

Profile velocity mode In profile velocity mode, the motor will rotate at a target velocity

Profile torque mode In profile torque mode, the drive sends a target torque to the motor

Homing modes

Diagnostics opmode Used for triggering diagnostic OS commands

Switching the modes Describes how the modes can be selected

Cyclic modes

- Cyclic synchronous torque control (CST) In torque control mode, the master sends desired torques to the drive
- Cyclic synchronous velocity control (CSV) In velocity control modes, the master sends desired velocities to the drive
- Cyclic synchronous position control (CSP) In position control modes, the master sends desired positions to the drive

In cyclic synchronous modes reference values are sent and updated with each EtherCAT communication frame, so a reference signal has to be sent with up to 4 kHz. As an example, to rotate the motor shaft from position A to position B in cyclic synchronous position mode within one second, the following information has to be sent to the drive:

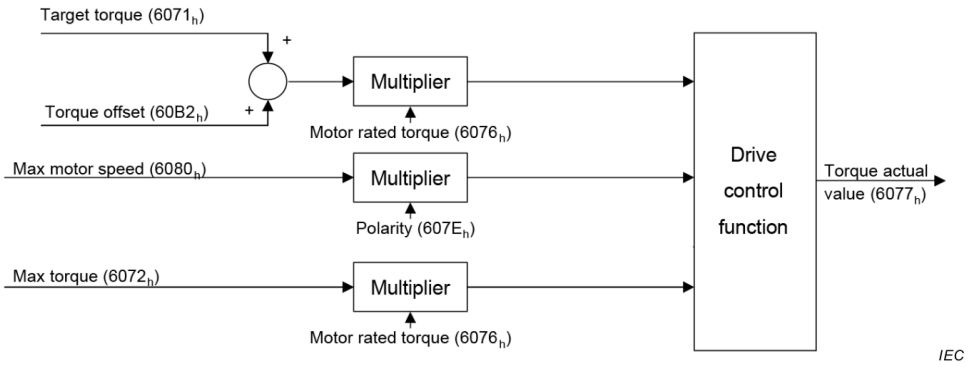
- position A in the beginning
- 3998 intermediate positions (ideally at 4 kHz)
- position B in the end

Cyclic Synchronous Torque Mode

In this mode, the trajectory generator is located in the control device, not in the servo drive. In a cyclic synchronous manner, it provides a target torque to the drive device, which performs torque control. Optionally, an additive torque value can be provided by the control system in order to allow two instances to set up the torque.

The cyclic synchronous torque mode covers the following sub-functions:

- demand value input;
- torque capture;
- torque control function with appropriate input and output signals;
- limitation of torque demand.



The figure above shows the inputs and outputs of the torque control function. The input (from the control function point of view) are the target torque and optionally a torque offset which is added to the target torque to allow two instances to set up the torque.

The torque can be limited in object 0x6072 Max torque.

The torque actual value is used as mandatory output to the control device.

Operation Mode dependent Bits of the Statusword

This mode uses some bits of the statusword for mode specific purposes which are indicated in yellow. The figure shows the structure of the statusword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-13	12	11	10	9-8	7-0
N/A	Target Torque ignored	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
  - Max torque 0x6072 limits Target torque 0x6071 and Torque offset 0x60B2.
- **Target reached:** The bit is set if the actual torque stays in the window of target torque ± torque window value for a duration of torque window time.
- **Target torque ignored:** This bit is set if Target torque 0x6071 is equal to Torque demand 0x6074.

  It means that the user command is passed directly through as an input of the torque controller. However, if Target torque is limited by Max torque, the limited value will be used as an input of the torque controller and the bit will be cleared.

  > Note:
  >
  > 0 = Target torque ignored
  >
  > 1 = Target torque shall be used as input

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x6071	Target torque	Use this object to directly select a Target torque
0x60B2	Torque Offset	This object provides the offset for the torque value

Cyclic Synchronous Velocity Mode

The velocity control modes allow the EtherCAT master to send desired velocities to the SOMANET drive. The drive will make sure these velocity values are reached by the motor (if the motor is physically able to do so).

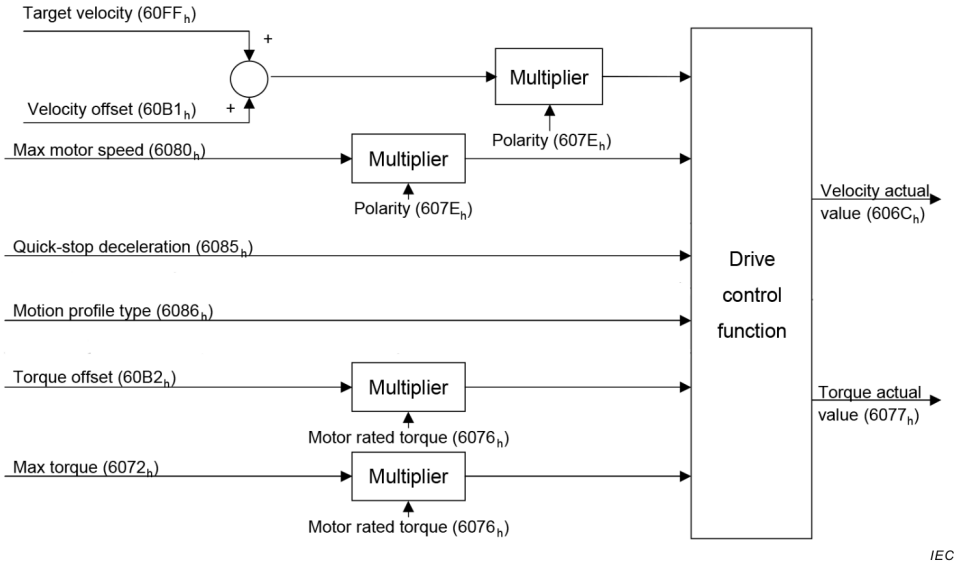
In this mode, the trajectory generator is located in the control device, not in the servo drive. In a cyclic synchronous manner, it provides a target velocity to the drive device, which performs velocity control and torque control. If desired, the position control loop may be closed over the communication system. Optionally, a velocity offset and a torque offset may be provided by the control system in order to allow a second source for velocity and/or a torque feed forward.

Position sensors are supported and the velocity may be calculated by taking the derivative.

The cyclic synchronous velocity mode covers the following sub-functions:

- demand value input;
- velocity capture using position sensor;
- velocity control function with appropriate input and output signals.

The behavior of the control function is influenced by control parameters such as limit functions, which are externally applicable.



The figure above shows the inputs and outputs of the drive control function. The input (from the control device point of view) are the target velocity and optionally, a velocity offset (to be added to the target velocity) as well as a torque offset. The drive control function in the diagram contains both the velocity controller and the torque controller, where the output of the velocity control loop is used as an input for the torque control loop.

The drive device supports limitation of motor speed and a quick stop function for emergency reasons.

The time period between two updates of the target velocity and/or additive velocity can be interpolated for smooth trajectories.

The velocity actual value is used as mandatory output to the control device. Another output is the torque actual value.

All values are in rpm by default and can be converted to user defined velocity units if needed.

Operation Mode dependent Bits of the Statusword

This mode uses some bits of the statusword for mode specific purposes which are indicated in yellow. The figure shows the structure of the statusword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-13	12	11	10	9-8	7-0
N/A	Target velocity ignored	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
  - Max. motor speed 0x6080 limits Target Velocity 0x60FF and Velocity Offset 0x60B1.
- **Target reached:** The bit is set if the actual velocity stays in the window of target velocity ± velocity window for a duration of Velocity window time.
- **Target velocity ignored:** This bit is set if Target velocity 0x60FF is equal to Velocity demand value 0x606B. It means that the user command is passed directly through as an input of the velocity PID controller.
- However, if Target velocity is limited by Max. motor speed, the limited value will be used as an input of the PID controller and the bit will be cleared.

> Note:
>
> 0 = Target velocity ignored
>
> 1 = Target velocity shall be used as input

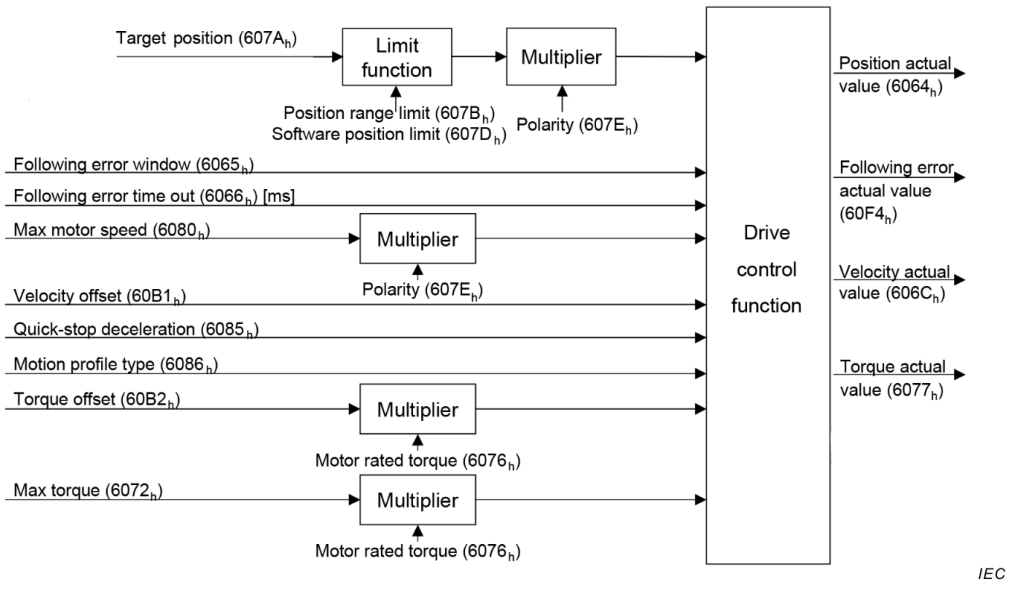
Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2011	Velocity Controller	Values Kp, Ki, Kd and Controller Integral Limit for Velocity Controller
0x60FF	Target velocity	Use this object to directly select a Target velocity

Cyclic Synchronous Position Mode

The position control modes allow the EtherCAT master to send desired positions to the servo drive. The drive will make sure these position values are reached by the motor (if the motor is physically able to do so).

In this mode, the trajectory generator is located in the control device, not in the servo drive. In a cyclic synchronous manner, it provides a target position to the drive device, which performs position control, velocity control and torque control.



The figure above shows the inputs and outputs of the drive control function. The input values (from the control function point of view) are the target position as well as an optional velocity offset and an optional torque offset used for feed forward control. Especially in a cascaded control structure, where a position control is followed by a velocity or torque control, the output of the position

control loop is used as an input for further calculation in the drive device. Limit functions can be used to restrict the range of values to avoid unintended positions. The drive device monitors the following error.

The target position is interpreted as absolute value. The position actual value is used as mandatory output to the control device. Further outputs are the velocity actual value and torque actual value. The following error actual value is used as an additional parameter. See Control supervision for details.

Three control loops are implemented for position control:

- Cascaded PID control
- Simple PID control
- Dual Encoder Cascaded Position Control

Operation Mode dependent Bits of the Statusword

This mode uses some bits of the statusword for mode specific purposes which are indicated in yellow. The figure shows the structure of the statusword. For the general structure and usage, please refer to our Application note on Status- and controlword.

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-13	12	11	10	9-0
N/A	Target position ignored	Internal limit active	Target reached	basic

- **Internal limit active:**
  - Max. motor speed 0x6080 limits Velocity demand value 0x606B and Profile velocity 0x6081.
  - Software position limit 0x607D limits Position demand value 0x6062.
- **Target reached:** the bit is set to 1 if the actual position stays in the window of the target position ± position window for a duration of position window time.

- **Target position ignored:** This bit is set if Target position (0x607a) is equal to Position demand internal value (0x60FC). It means that the user command is passed directly through as an input of the position PID controller.

  However, if the Target position is limited by Software position limits, the limited value will be used as an input of the PID controller and the bit will be cleared.

> Note:
>
> 0 = Target position ignored
>
> 1 = Target position shall be used as input

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)
0x607A	Target position	Use this object to directly select a Target position
0x60B1	Velocity offset	Sets a feed-forward velocity as input to the velocity controller
0x60B2	Torque offset	Sets a feed-forward torque as input to the velocity controller

Profile modes

SOMANET allows to generate a linear profile for position and velocity loops.

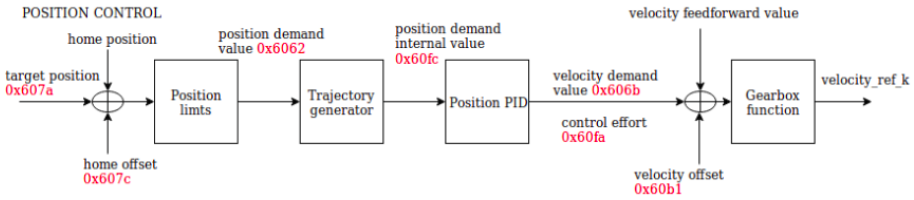
Profile position mode In profile position mode, the trajectory to a target point is generated on the drive

Profile velocity mode In profile velocity mode, the motor will rotate at a target velocity

Profile torque mode In profile torque mode, the drive sends a target torque to the motor

Profile position mode

In Profile Position Mode a target position is applied to the trajectory generator. It is generating a position demand value for the position control loop taking into account the position limits. The trajectory generator computes the position demand internal value depending on the configured speed, starting acceleration and braking deceleration.



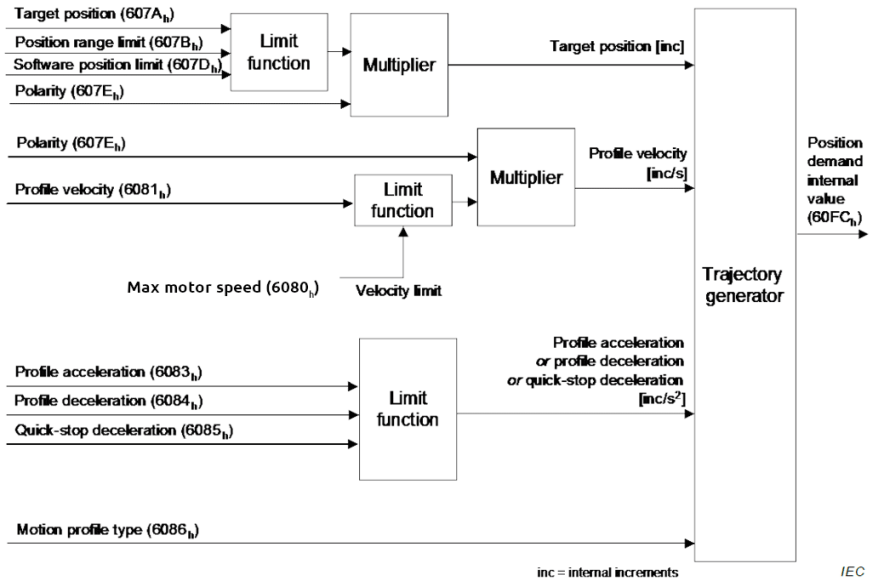
Objects configuration

The following objects need to be configured for the position profile:

- 0x6081 - Profile Velocity
- 0x6083 - Profile acceleration
- 0x6084 - Profile deceleration
- 0x6085 - Quick Stop deceleration

> Note:
>
> Profile velocity is limited by the Object:
>
> Max motor speed 0x6080

Trajectory generator

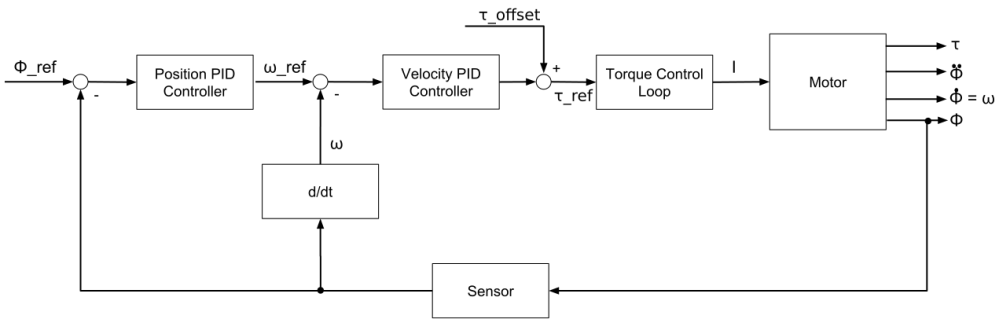


The trajectory generator receives the Target position 0x607A, taking into account the Position range limit 0x607B, the Software position limit 0x607D, as well as the Polarity 0x607E.

> Note:
>
> Target position is limited by the Objects:
>
> - Position range limit 0x607B
>
> - Software position limit 0x607D

The output of the trajectory generator is the position demand internal value for the position PID.

General view



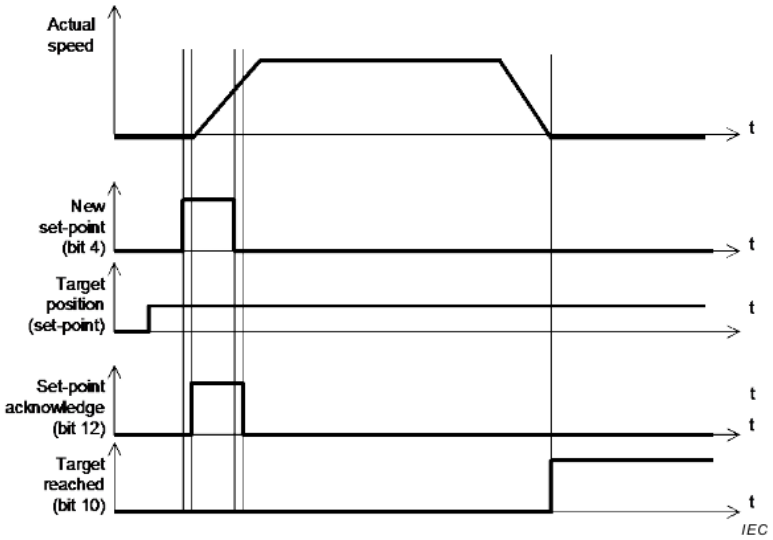
Enable Profile position mode

To enable the mode, the value 1 (0001h) must be entered in object Modes of operation 0x6060. The object Op mode display 0x6061 can be used to check if the op mode has been set correctly.

Setting set-points

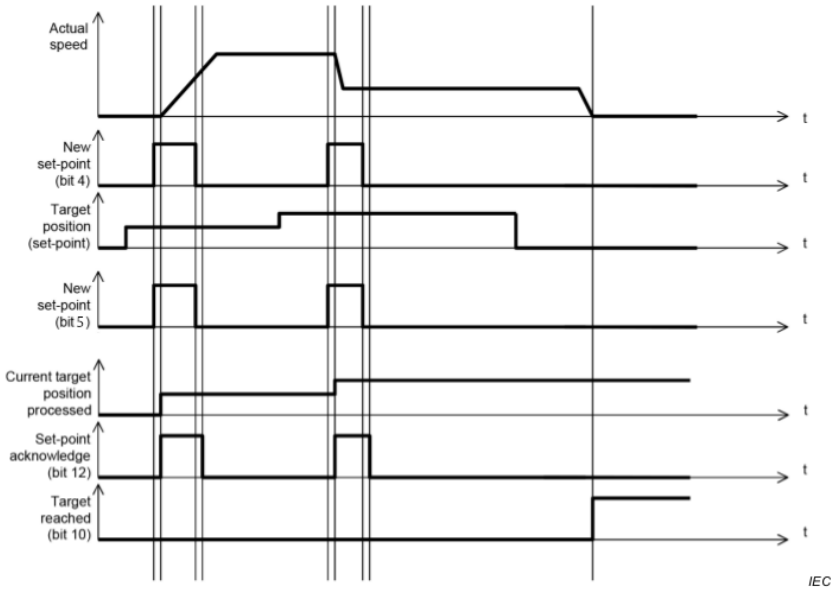
The setting of set-points is controlled by the timing of the *new set-point bit* in the controlword as well as the *set-point acknowledge bit* in the statusword.

After a set-point is applied to the drive device, the control device signals that the set-point is valid by a rising the edge of the *new set-point bit* in the controlword. The drive device sets the *set-point acknowledge bit* in the statusword to 1, afterwards the drive device signals with the *set-point acknowledge bit* set to 0 its ability to accept new set-points.



In order to execute the new set-point (new target value), bit 4 (*new set-point*) of the controlword needs to be changed from 0 to 1 (the profile reacts on the rising edge of the bit). The drive will respond with the rising edge of bit 12 (*set-point acknowledge*) of the statusword. The Single set-point method is used, which means that bit 5 of the controlword (*Change set immediately*) has to be set to 1.

> Note:
>
> The drive won't wait for the previous set-point to finish.
>
> To execute the new set-point during the execution of the previous one, the **rising edge of the `new set-point` bit needs to be generated again.**



Operation Mode dependent Bits of Status- and Controlword

Controlword

The profile position mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

15-9	8	7-6	5	4	3-0
N/A	Halt	N/A	Change set immediately	New set-point	basic

- **New set-point:** On the rising edge, the drive will acknowledge the value of “target position” as a new set-point.
- **Change set immediately**: Our drive supports only a single-set-point method of handling. Therefore, this bit has to be 1 before setting a new-point.
- The new set-points will be executed immediately and won’t wait for the previous one to finish.

Attention:

This **bit 5 has to be set before or simultaneously with bit 4 `New set point`**. If this bit is not set, the profile command will not be executed!

- **Halt**: When the bit is activated, the drive will use the profile deceleration to put the motor's shaft into a standstill position.

Note:

The Halt bit is used to toggle the motion. When it is set to 1, the profile deceleration ramp will be used to reach 0 velocity. If it is set to 0 again, the demanded operation will be continued and the drive will accelerate using the profile acceleration ramp until the desired position has been reached.

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

Objects related to bits 12 and 13 of the statusword:

- 0x6065 - Following error window
- 0x6066 - Following error timeout
- 0x6067 - Position window
- 0x6068 - Position window time

15-14	13	12	11	10	9-8	7-0
N/A	Following error	Set-point acknowledge	Internal limit act-	Target reached	N/A	basic

15-14	13	12	11	10	9-8	7-0
			ive			

- **Internal limit active:**
  - Max. motor speed 0x6080 limits Velocity demand value 0x606B and Profile velocity 0x6081
  - Software position limit 0x607D limits Position demand value 0x6062.
  - Max torque 0x6072 limits Torque demand 0x6074.
- **Target reached:** the bit is set to 1 if the actual position stays in the window of the target position ± position window for a duration of position window time.
- **Set-point acknowledge:** Is set to 1 when the drive acknowledges the rising edge on the *new-set point bit* of the controlword.
- **Following error**: the bit is set to 1 If the actual position stays in the window of position demand value ± following error window for a duration of following error time out.
- For more details check the object Following error actual value 0x60F4.

> Note:
>
> If the position window is 0xFFFFFFFF position window monitoring will be switched off and the target reached bit will always be 0.

Example: Using the position profile

- set the profile objects:
  - Profile Velocity 0x6081
  - Profile acceleration 0x6083
  - Profile deceleration 0x6084
  - Quick Stop deceleration 0x6085
- set mode of operation to 1

- Set target position
- Switch the drive to “Operation Enabled”
- Set bit 5 of the controlword:
- Set bit 4 of the controlword:
    - The drive will acknowledge the new target position on the rising edge of bit 4 by activating the statusword bit 12 *Set-point acknowledge* for several cycles.
    - The motor will start rotating immediately.

> Note:
> Setting the bits 4 and 5 of the controlword has to be done in the same order shown above or simultaneously.

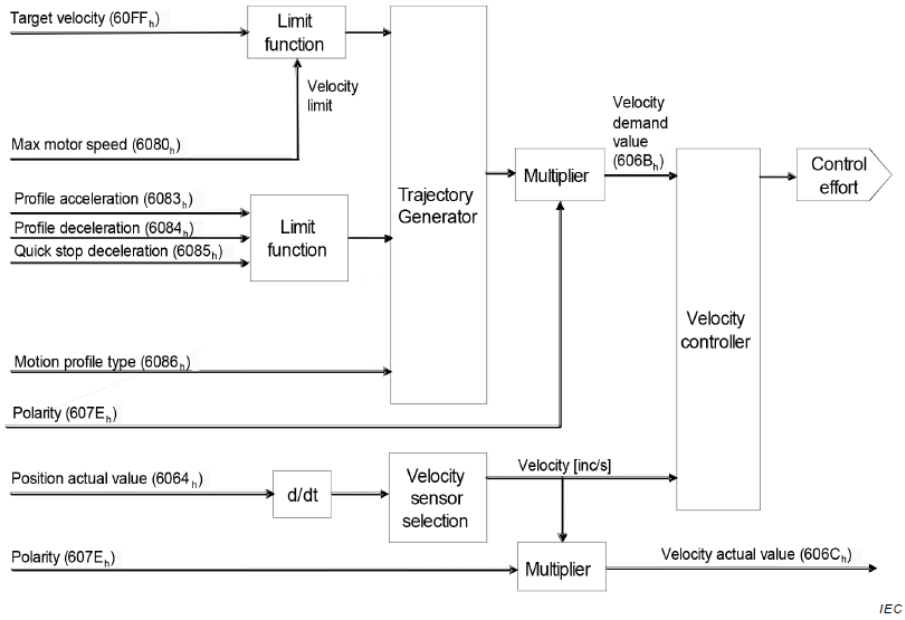
Parameters

Object ID	Name	Description
0x6081	Profile Velocity	
0x6083	Profile acceleration	
0x6084	Profile deceleration	
0x6085	Quick Stop deceleration	
0x6080	Max motor speed	The maximum possible speed of the Motor according to the data sheet
0x6072	Max torque	
0x6074	Torque demand	

Object ID	Name	Description
0x607A	Target position	Use this object to directly select a Target position
0x607B	Position range limit	
0x607D	Software position limit	
0x607E	Polarity	
0x6060	Modes of operation	
0x6061	Op mode display	
0x6065	Following error window	
0x6066	Following error timeout	
0x60F4	Following error actual value	
0x6067	Position window	
0x6068	Position window time	
0x606B	Velocity demand value	
0x6062	Position demand value	

Profile velocity mode

In profile velocity mode the drive accelerates until the demand velocity value has been reached and keeps this velocity.



Objects configuration

The following objects need to be configured for the velocity profile:

- 0x60FF - Target velocity
- 0x6083 - Profile acceleration
- 0x6084 - Profile deceleration
- 0x6085 - Quick stop deceleration
- 0x607E * - Polarity
- 0x6080 - Max motor speed

* This object is automatically detected and configured by Commutation Offset Detection.

> Note:
>
> Target velocity is limited by Object:
>
> Max motor speed 0x6080

The motion in Profile velocity mode is toggled by the halt bit (8) in the controlword (object 0x6040):

- Value 0: Motion is executed, the drives accelerates (with the specified Profile acceleration) until either
    1. the velocity is reached or
    2. halt bit is set to 1
- Value 1: Motion is stopped, the drive decelerates (with the specified Profile deceleration) until either
    1. velocity = 0 or
    2. halt bit is reset to 0

> Attention:
>
> When the drive mode is set to Profile velocity mode and the halt bit is unset, the motor will start turning immediately.

Enable Profile velocity mode

To enable the mode, the value 3 (0003h) must be entered in object 0x6060. The Op mode display 0x6061 can be used to check if the op mode has been set correctly.

Operation Mode dependent Bits of Status- and Controlword

Controlword

The profile velocity mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

15-9	8	7-4	3-0
N/A	Halt	N/A	basic

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

Objects related to bits 10 and 12 of the statusword:

- 0x606D - Velocity window
- 0x606E - Velocity window time
- 0x606F - Velocity threshold
- 0x6070 - Velocity threshold time

5-13	12	11	10	9-8	7-0
N/A	Speed bit	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
  - Max. motor speed 0x6080 limits Velocity demand value 0x606B and Profile velocity 0x6081.
  - Max torque 0x6072 limits Torque demand 0x6074.
- **Target reached:** The bit is set if the actual velocity stays in the window of target velocity ± velocity window for a duration of *Velocity window time*.
- **Speedbit:**The speed bit is set to 1 when the actual velocity goes below the velocity threshold (meaning the motor is in standstill).

The bit will be 0 when the actual velocity exceeds the velocity threshold of the speedbit for a duration of *Velocity threshold time* (meaning the motor is rotating).

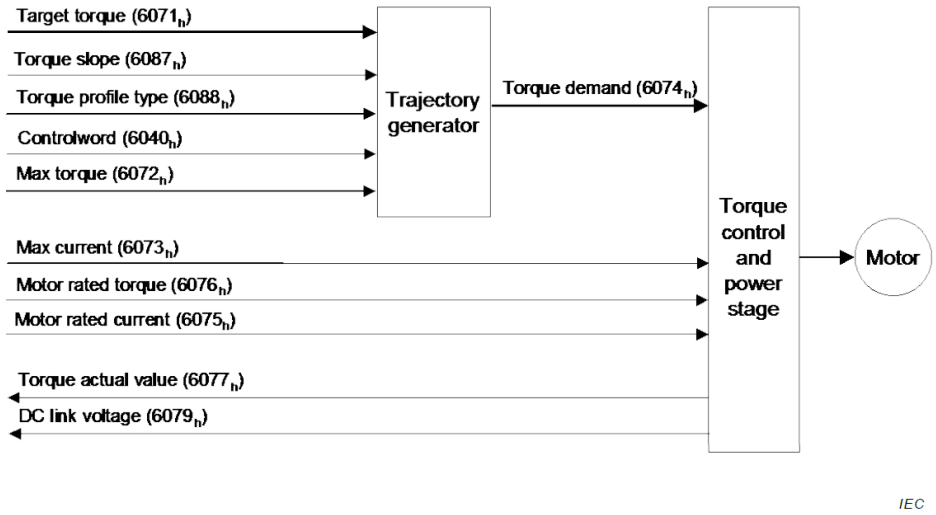
> Note:
>
> If the velocity window is 0xFFFF, velocity window monitoring will be switched off and the target reached bit will always be 0.

Parameters

Object ID	Name	Description
0x60FF	Target velocity	
0x6083	Profile acceleration	
0x6084	Profile deceleration	
0x6085	Quick stop deceleration	
0x607E	Polarity	
0x6080	Max motor speed	
0x606D	Velocity window	
0x606E	Velocity window time	
0x606F	Velocity threshold	
0x6070	Velocity threshold time	
0x6080	Max. motor speed	
0x606B	Velocity demand value	
0x6081	Profile velocity	
0x6072	Max torque	
0x6074	Torque demand	

Profile torque mode

In profile torque mode, a constant torque is generated according to the target torque and the defined torque slope (the rate of change of torque).



Output of the generator is the torque demand value (0x6074) which is processed by the torque controller.

Objects configuration

The following objects need to be configured for the torque profile:

- 0x6071 - Target torque
- 0x6087 - Torque slope
- 0x6085 - Quick Stop deceleration

> Note:
>
> 0x6071 Target torque is limited by the Object :
>
> Max torque 0x6072

Enable Profile torque mode

To enable the mode, the value 4 (0004h) must be entered in object 0x6060. The Op mode display 0x6061 can be used to check if the op mode has been set correctly.

> Note:
>
> Both enabling the halt bit and setting the target torque to zero will ramp down the torque applied to the motor according to the torque slope. At the end of the slope no torque will be

pplied to the motor, allowing the shaft to move freely.

Starting/Stopping Torque

To start and stop motion, toggle the controlword halt bit (bit 8). When the halt bit is set to 0, motion will start or continue; when the halt bit is set to 1, motion will stop. The bit can be toggled by writing 010Fh and 000Fh to controlword (object 0x6040).

Operation Mode dependent Bits of Status- and Controlword

Controlword

The torque position mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

15-9	8	7-4	3-0
N/A	Halt	N/A	basic

- **Halt:** Both enabling the halt bit or setting the target torque to zero will ramp down the torque applied to the motor according to the torque slope object (0x6087). At the end of the slope, no torque will be applied to the motor, allowing the shaft to move freely.

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-12	11	10	9-8	7-0
N/A	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
    - Max torque 0x6072 limits Torque demand 0x6074, Target torque 0x6071 and Torque offset 0x60B2
- **Target reached:** the bit is set if the actual torque stays in the window of target torque ± torque window value for a duration of torque window time.

Objects related to the bit 10 of the statusword:

Torque window value 0x606D Torque window time 0x606E

Parameters

Object ID	Name	Description
0x6071	Target torque	
0x6087	Torque slope	
0x6085	Quick Stop deceleration	
0x6072	Max torque	
0x6074	Torque demand	
0x60B2	Torque offset	
0x606D	Torque window value	
0x606E	Torque window time	

Homing modes

Overview

Axes without an absolute encoder can use the homing function to restore the position to a previously defined location.

Homing methods

SOMANET supports the following methods as defined in CiA 402 profile specification (IEC61800-7-201:2015):

Method Number	Description
1	Homing on a negative limit switch and index pulse
2	Homing on a positive limit switch and index pulse
3	Homing on a positive home switch inactive and index pulse
4	Homing on a positive home switch active and index pulse
5	Homing on a negative home switch inactive and index pulse
6	Homing on a negative home switch active and index pulse
7	Homing on a home switch (inactive) and index pulse with positive initial direction. Reverse if positive limit switch is hit
8	Homing on a home switch (active) and index pulse with positive initial direction. Reverse if positive limit switch is hit
9	Homing on an opposite home switch (active) and index pulse with positive initial direction. Reverse if positive limit switch is hit
10	Homing on an opposite home switch (inactive) and index pulse with positive initial direction. Reverse if positive limit switch is hit
11	Homing on a home switch (inactive) and index pulse with negative initial direction. Reverse if negative limit switch is hit
12	Homing on a home switch (active) and index pulse with negative initial direction. Reverse if negative limit switch is hit
13	Homing on an opposite home switch (active) and index pulse with negative initial direction. Reverse if negative limit switch is hit
14	Homing on an opposite home switch (inactive) and index pulse with negative initial direction. Reverse if negative limit switch is hit

Method Number	Description
33	Homing on an index pulse with negative initial direction
34	Homing on an index pulse with positive initial direction
35/37 (identical)	Homing on the current position

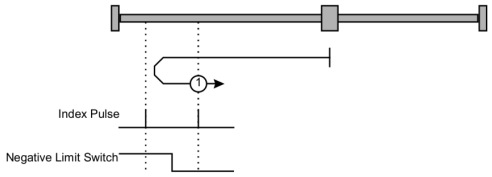
Homing methods 1-14 (Homing on limit switches)

In all switch based homing methods the axis moves to search the switch (Home Switch or Limit Switch).

The following images represent the direction of movement and the starting position of each homing method.

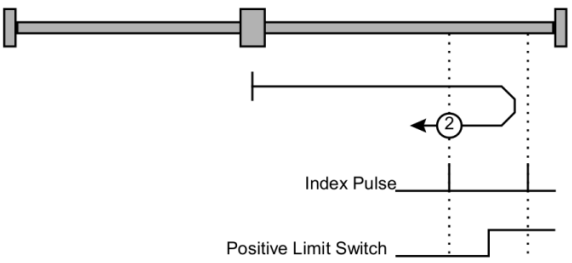
The respective method is indicated by the encircled number.

Homing Method 1 : Homing on a Negative Limit Switch and Index Pulse



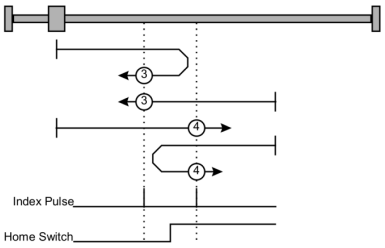
- The initial direction of movement shall be leftward if the negative limit switch is inactive.
- The home position is the first index to the right when the negative limit switch changes state.

Homing Method 2 : Homing on a Positive Limit Switch and Index Pulse



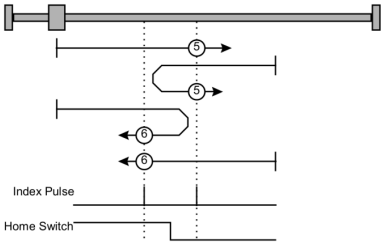
- The initial direction of movement shall be rightward if the positive limit switch is inactive.
- The home position is the first index to the left when the positive limit switch changes state.

Homing Method 3,4 : Homing on a Positive Home Switch and Index Pulse



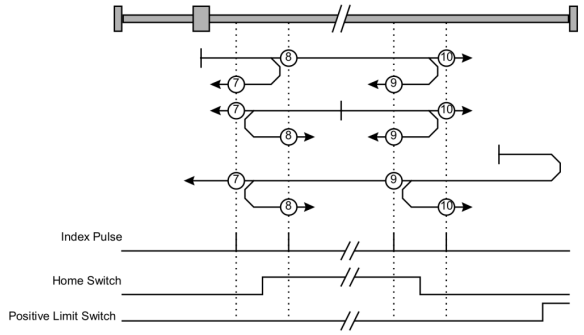
- The initial direction of movement is dependent on the state of the home switch: leftward if positive home switch active and rightward if it’s inactive.
- Method 3: the home position is the first index to the left when the positive home switch changes state (home switch inactive).
- Method 4: the home position is the first index to the right when the positive home switch changes state (home switch active).

Homing Method 5,6 : Homing on a Negative Home Switch and Index Pulse



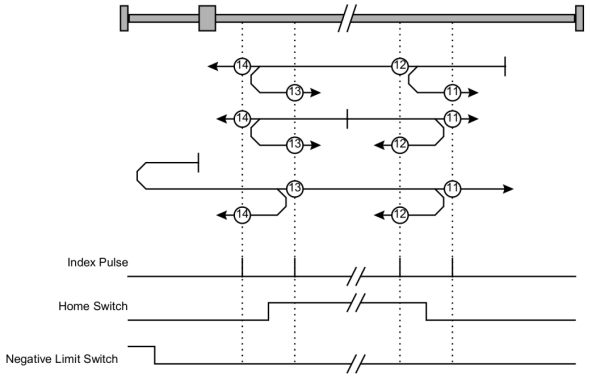
- The initial direction of movement is dependent on the state of the home switch: leftward if negative home switch inactive and rightward if it’s active.
- Method 5: the home position is the first index to the right when the negative home switch changes state (home switch inactive).
- Method 6: the home position is the first index to the left when the negative home switch changes state (home switch active).

Homing Method 7 to 10 : Homing on a Home Switch and Index Pulse , Positive Limit Switch Required to revert direction



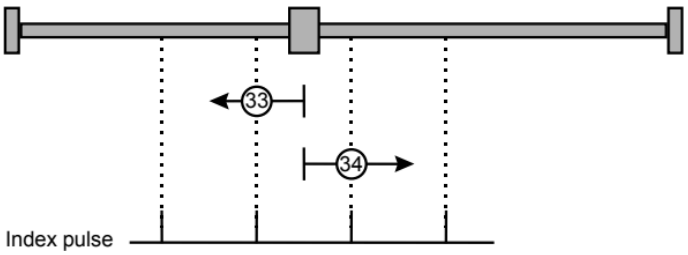
- The initial direction of the movement is rightward except if home switch is initially active.
- If the home switch is initially active then the direction of movement depends on the homing method.
- The direction will be reversed when encountering the positive limit switch.

Homing Method 11 to 14 : Homing on a Home Switch and Index Pulse , Negative Limit Switch Required to revert direction



- The initial direction of the movement is leftward except if home switch is initially active.
- The direction will be reversed when encountering the negative limit switch.
- If the home switch is initially active then the direction of movement depends on the homing method.

Homing method 33/34 (move to next or previous single-turn position or “index pulse”)



- Methods 33 and 34 can be used to center on the sensor’s reference position (in positive or negative direction).
- The shaft will rotate less than one rotation.

Homing method 35/37 (zero the current position)

- Methods 35 and 37 are used to define the current position as home position.

Note:

Homing method 35/37 can be performed in any operational state. All other homing methods require the drive to be in “operation enabled” state.

Use case:

Homing on current position while operation is not enabled

If the axis is moved to a desired homing position by external force (for example manually),

Homing Mode 35/37 can be used to determine the current position as home.

Use case:

Relative position command

The current position can be taken as the reference point.

Use case:

Homing on current position

When the motor is moved to the desired position in CSP the position can be taken as homing position.

Note:

Homing method 35/37 can be performed in any state, as the motor is not controlled there.

All other homing methods require the drive to be in “operation enabled” state.

Settings for homing mode

State for homing modes

All homing methods except for methods 35/37 require the drive to be in the “operation enabled” state. After the home position is found, the position controller stays active, holding the motor at that position (except for method 35/37).

Homing method

You can select the desired homing method in object 0x6098: Homing Method.

Mode	Description
0	No homing method assigned (Default)
1	Homing on negative home switch and index pulse
2	Homing on positive home switch and index pulse
3,4	Homing on positive home switch and index pulse
5,6	Homing on negative home switch and index pulse
7 to 14	Homing on the home switch with an index pulse
17, 18	Similar to method 1 & 2, but without index pulse. Homing is done at switch

Mode	Description
	transition
19 to 22	Similar to methods 3 to 6, but without index pulse. Homing is done at switch transition
23 to 30	Similar to method 7 to 14, but without index pulse. Homing is done at switch transition
33	Homing on negative index pulse
34	Homing on positive index pulse
35	Homing on current position. (deprecated, please use method 37)
37	Homing on current position. (identical with method 35)

Homing speeds

Two homing speeds must be selected in object 0x6099: Homing Speed: in a typical cycle, the faster speed is used to find the home switch and the slower speed is used to find the index pulse.

> Note:
>
> Homing speeds are not required in method 35/37.

Homing acceleration

In object 0x609A: Homing Acceleration a value for acceleration during homing must be defined.

> Note:
>
> An acceleration is not needed in method 35/37.

Setting digital I/Os for Homing

Object 0x2210 (GPIO Pin configuration) is used to configure the digital inputs for homing and limit switches. Switches can be configured in both active high and active low configuration. But only one GPIO can be assigned to one switch whether active high or active low. For example If Pin 3 is

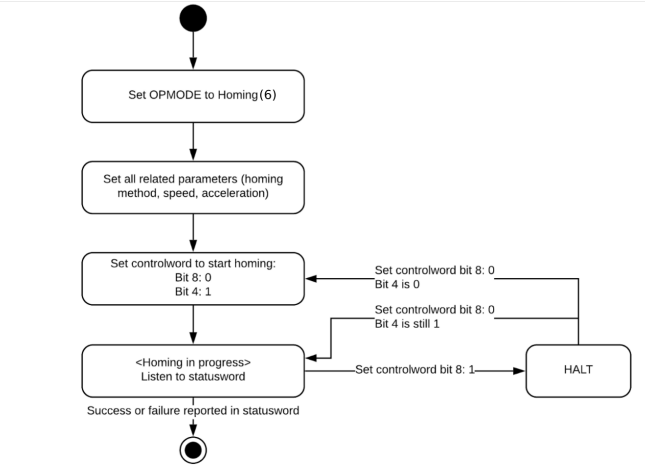
configured as Active High Limit Switch, then configuring any other pin as Limit Switch Active High or Active Low will raise the error IvldGpio (Invalid GPIO error) in the error report object.

Further details about configuring digital I/Os can be found in Digital GPIO

Object 0x60FD (Digital inputs) reports the state of the digital inputs which can be configured for limit switches, interlock, etc.

How to perform homing

The following diagram shows the steps that are necessary to perform the homing function.



After the home position is found, the position controller stays active, holding the motor at that position (except for method 35/37).

Using the Halt bit

The halt bit (bit 8 of the Controlword 0x6040 can be used to stop the drive without disabling the controller:

- Set the halt bit to 1 to stop the motion
- When the halt bit is reset to 0, the motion continues

Status of the homing process

Bits 10, 12, 13 of the Statusword 0x6041 are used to indicate the current state of the homing process:

- Bit 10: Target reached
- Bit 12: Homing attained
- Bit 13: Homing error

Bit 13	Bit 12	Bit 10	Status
0	0	0	Homing is performed
0	0	1	Homing is interrupted or not started
0	1	0	Homing confirmed, but target not yet reached
0	1	1	Homing completed
1	0	0	Error detected, motor still turning
1	0	1	Error during homing, motor at standstill

Hitting a limit switch in homing mode

Homing modes are meant to be used with homing switches whereas limit switches are considered as physical end points.

When a limit switch is activated in a homing mode other than method 1-14 (which rely on limit switches), this is considered a fault and the drive transitions to “Drive disabled” state.

Operation Mode dependent Bits of Status- and Controlword

Controlword

Homing mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

15-9	8	7-5	4	3-0
N/A	Halt	N/A	Homing operation start	basic

- **Halt:** If halt is triggered (Bit 8 of control word is set as 1) during homing, independant of the step the homing routine is in, Axis begins to ramp down to go to zero velocity. The axis stays at zero velocity as long as the halt remains active. When the halt bit is lifted, the homing routine is resumed from the same step it was in before halt.

- **Homing operation start:** set the homing operation start bit to 1 to start the homing procedure.

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information, please refer to the section on Control Supervision.

15-14	13	12	11	10	9-8	7-0
N/A	Homing error	Homing attained	Internal limit active	Target reached	N/A	basic

Parameters

Object ID	Name	Description
0x6098	Homing Method	Specifies the homing method that shall be used
0x607C	Home Offset	Offset from user's zero position to the home position
0x6099	Homing Speeds	Speed with which to move during homing
0x609A	Homing Acceleration	Acceleration used during homing
0x6040	Controlword	Set bit 8 to 0 and bit 4 to 1 to start homing
0x6041	Statusword	Indicates the current state of the homing process
0x60FD	Digital inputs	Indicates the current state of the digital inputs

Object ID	Name	Description
0x2210	GPIO Pin configuration	Configures the digital inputs for homing and limit switches
0x2005	Homing	Options configures different options for homing modes

Note:

Switches can be configured in both active high and active low configuration. But only one GPIO can be assigned to one switch, whether active high or active low. For example, if Pin 3 is configured as Active High Limit Switch, then configuring any other pin as Limit Switch Active High or Active Low will raise IvldGpio (Invalid GPIO error).

Diagnostics opmode

This operational mode allows triggering diagnostics OS commands. The following OS commands can only be executed when the drive is in this mode and when the CiA402 state is Operation enabled:

- Command 4: Motor phase order detection
- Command 5: Commutation offset detection
- Command 6: Open phase detection
- Command 7: Pole pair detection
- Command 8: Phase resistance measurement
- Command 9: Phase inductance measurement
- Command 10: Torque constant measurement

To select Diagnostics opmode, the value of -2 must be written to the object 0x6060 Modes of operation. When this opmode is selected and the transition to Operation enabled is performed, the brake will only be disengaged if the encoder index needs to be found. Once this procedure is done, the brake will be engaged again and object 0x6041 Statusword will reflect that the drive is in Operation enabled state. When no OS command is running, the motor control is disabled.

Attention:

Always ensure that the brake is engaged or disengaged depending on the command that is executed:

Command 4 rotates the rotor

Command 5 rotates the rotor when the selected method is 0 or 1

Command 6 does not need to rotate the rotor. In case a brake is attached, it may stay engaged during this command

Command 7 rotates the rotor

Command 8 rotates the rotor

Command 9 rotates the rotor

Command 10 rotates the rotor

For example, before running a diagnostics OS command that requires movement, the user should disengage the brake. When the OS command is finished the user should engage the brake again, so that the load does not fall. An exception to this would be if the user is planning to execute another diagnostics OS command that requires movement, because in that case it could run it right after, so the motor control would be disabled just for a few milliseconds.

Selecting and switching the Modes

Selection of these modes is done by setting the object 0x6060 to the corresponding value as defined in the CiA 402 standard.

The modes are called by setting the object 0x6060 as follows:

Value for 0x6060	Control Mode
+1	Profile position mode
+3	Profile velocity mode
+4	Torque profile mode
+6	Homing mode

Value for 0x6060	Control Mode
+8	Cyclic Syncronous Position Mode
+9	Cyclic Syncronous Velocity Mode
+10	Cyclic Syncronous Torque Mode

Dynamic Op-Mode switching

The modes can be switched while in state *Operation enabled*.

> Attention:
>
> Ensure that all required PDO data is available for the mode you are switching to.

Object ID	Name	Description
0x6060	Op Mode	Use this object to select the desired Mode

Extended Control Functionalities

Synapticon provides several features for enhancing motion control even with low-cost hardware:

Cogging compensation Removes ripples caused by magnetic interaction between motor rotor and stator

Field weakening The velocity range of electric motors can be extended by weakening the magnetic field of the rotor linearly over the speed range

Anti-windup control An Anti-Windup algorithm is used to prevent the Integrator part to accumulate beyond acceptable boundaries

Gain scheduling Applies controller gains, which will change dynamically regarding the current speed

Motor overload protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating

Velocity feed forward To increase the velocity of the robot's reaction, the target position signal can bypass the position controller

Torque offset A desired offset torque can be added to the reference torque

Velocity offset An additional velocity offset can be used to improve the positioning performance

Filtering Digital filters can be used to remove unwanted frequencies such as noise or nonlinear effects

Position and Velocity Feedback Low-pass filters

Torque loop input shaping filter

Command smoothing and Interpolation Successive setpoints can be smoothed, so that low command resolution and jitter have a reduced effect on the control

Control supervision Allows to supervise the work of the control loops

Distributed clocks (DC clocks) Allows to synchronize the system time for enhanced performance

OS Command Offers commands that can't be executed by single SDO operations

Quick stop Stops the motor quickly in a controlled way

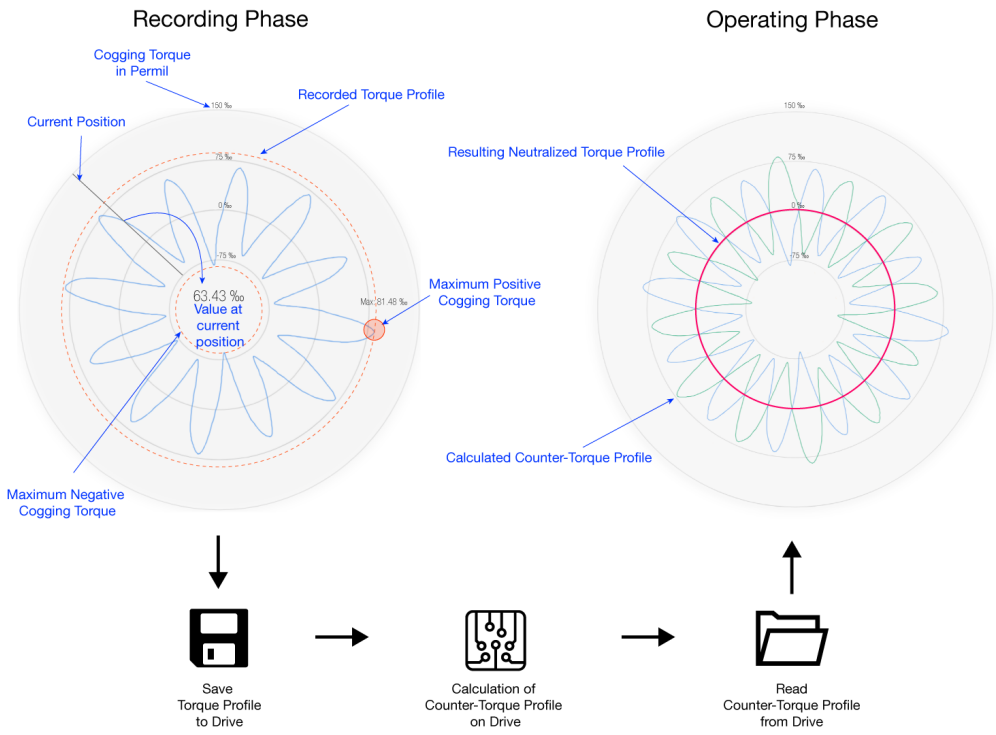
Cogging Torque Compensation

Overview

Cogging is an effect caused by magnetic interaction between the motor rotor and the stator. It results in torque ripples over the motor revolution, which leads to uneven motion or generally unsatisfying performance of the motor at low velocities. This phenomenon is of particular concern in low-cost motor models but can be compensated on a software level.

Cogging torque compensation is a software feature that suppresses torque ripple of electric motors to overcome vibration problems in high precision and low velocity applications. It can be found in OBLAC Box.

When cogging torque compensation is activated, an additional torque component is generated actively by the motor in order to compensate for the cogging torque. The compensation feature makes use of the fact that cogging torque is dependent on the single-turn motor position. To find the amount of compensation torque needed at each motor position, a recording procedure has to be performed before the compensation itself can be used.



Attention:

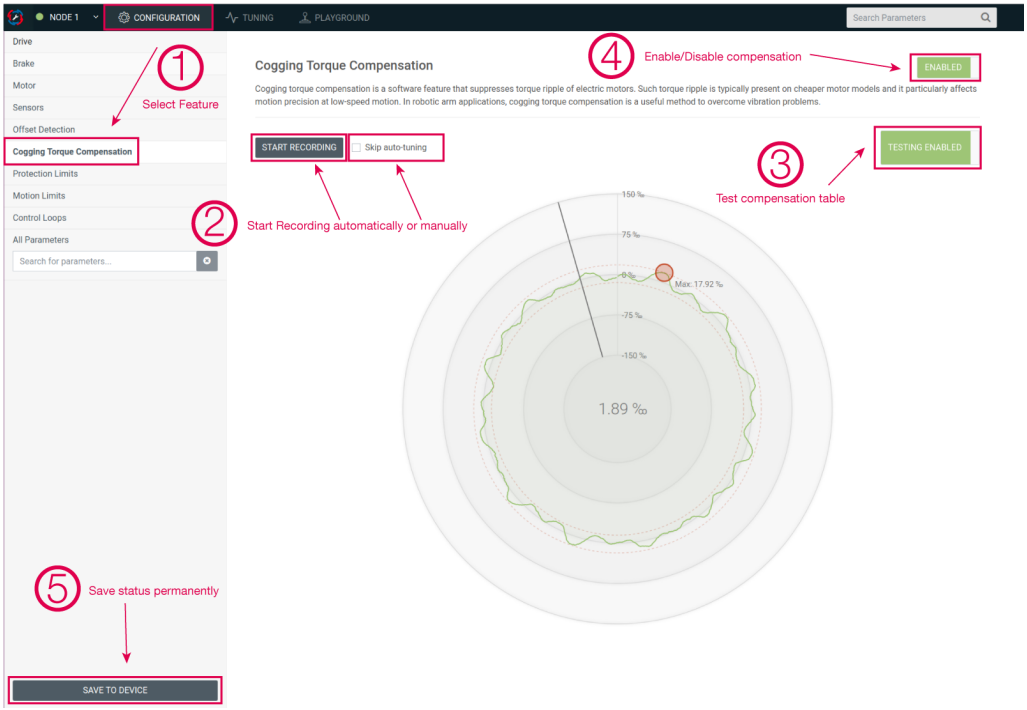
During the recording phase, the motor will do several turns at low velocities. Since cogging is a property of the motor and independent of attached loads it's possible to perform the procedure outside the actual setup.

Procedure

Prerequisites

- The minimum resolution of the encoder required is 1024 counts per revolution (cpr), it is recommended to use a resolution of 4096 cpr or higher.
- The motor and the drive should be properly sized. In particular, the motor's rated current should be at least 20 % - 30 % of the drive's rated current.
- For executing the cogging torque recording the motor must not be mechanically blocked and ideally must have no load attached.
- The motor is fully functional and the drive is properly configured.
- The System identification and position auto-tuning procedures were successfully executed with the current setup. The position control mode is fully functional.

Overview of the GUI



Recording

Before recording the cogging torque compensation table, a special configuration of the drive is required. This is done automatically, but can also be executed manually. When the recording is finished, this configuration is discarded and the old values are restored.

Automatic recording

This is the recommended default option that will do the procedure automatically. It uses a mathematical model of the electromechanical system obtained during the system identification to tune the controller. This procedure might take up to 3 minutes.

After setting up the drive and successfully executing the system identification, enter the menu item “Configuration” and select “Cogging Torque Compensation”.

To use the automatic recording, simply click on “Start Recording”

> Attention:
>
> During the recording routine the motor will do several turns.

Testing the compensation table

Once the recording is finished, the values are stored on the Drive. In the OBLAC Box GUI the feature is disabled and can now be tested manually:

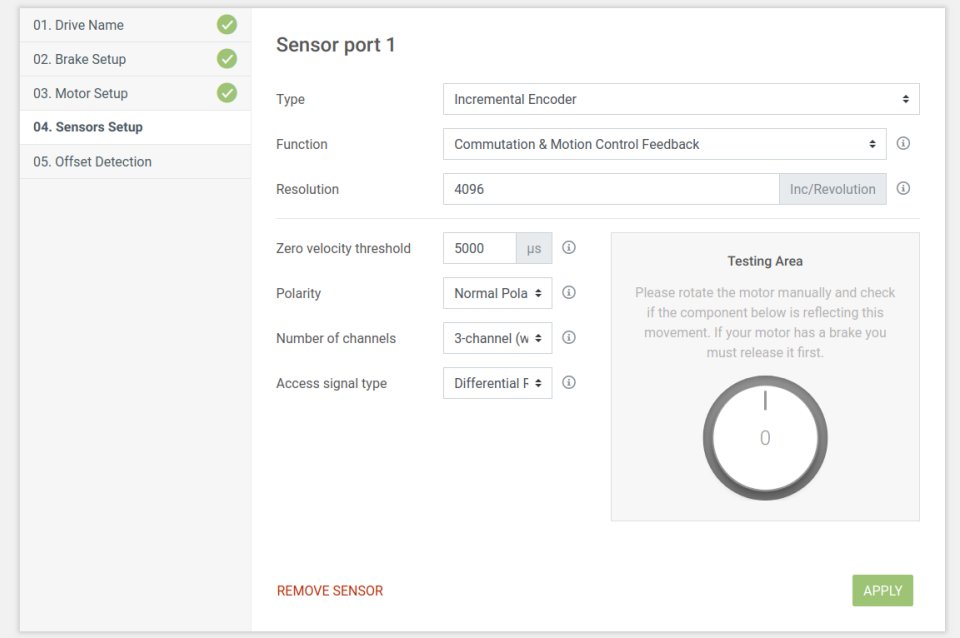
1. Slide the switch to TESTING ENABLED
2. Turn the shaft manually, you will feel the torque ripples.
3. Set the cogging torque compensation feature to ENABLED
4. Turn the shaft again manually and feel the difference

Manual configuration for the cogging recording

This option should be used when the performance of the auto-tuning is not satisfactory. In this case, the user should manually configure and tune the system. When the cogging torque compensation recording is triggered, the system will just use the current configuration.

Manually preparing the system for the cogging torque compensation recording:

1. An encoder on the motor shaft has to be used for the cogging torque compensation table recording. Set the function of this encoder to: "Torque", "Velocity" and "Position". This is done in the setup wizard.



2. Go to the tuning section and perform System identification.
3. Select “Position control mode” and tune the cascaded control loop with the sliders.
4. The tuning should be as sharp as possible: try setting the settling time slider to 50-80 ms.
5. Verify the tuning by setting the position ramp profile to 10 RPM velocity. The actual velocity should be as close to 10 RPM as possible.
6. Go to the cogging torque compensation screen and check the ‘Skip auto-tuning’ checkbox.
7. After the recording is finished, test the obtained compensation table:

Use switch ‘TESTING DISABLED’ in the cogging torque compensation screen to enable the torque control. Switch ‘DISABLED/ENABLED’ to enable/disable the cogging torque compensation table and feel the difference by rotating the motor shaft by hand.

8. Restore original configuration of the drive and re-tune control loops if possible.

Using the compensation table

Once the recording procedure is successfully finished, the cogging torque compensation can be activated and deactivated both in the OBLAC Drives GUI as well as via EtherCAT. The Object 0x2008 ”Cogging torque compensation is set when the cogging torque compensation is enabled or disabled in the GUI of OBLAC Box .

Set the switch for the cogging torque compensation to “Enabled” and click “Save to Device” to use the feature during operation of the drive.

The feature is activated globally and will be used in all modes of operation (torque control, velocity control and position control).

Controlling the feature via EtherCAT

To enabled or disabled the compensation algorithm Object 0x2008 ”Cogging torque compensation”, subindex 2 (cogging torque compensation enabled) must be set to 1.

By enabling the cogging torque compensation algorithm, the table communicated to the torque controller is added as a set of values to the torque reference value depending on the motor position. If the algorithm is disabled, this is skipped.

To start the cogging torque compensation recording procedure via EtherCAT, the manufacturer specific operational mode “COGGING COMPENSATION RECORDING” must be enabled in Object 0x6060 (Op Mode) by setting the value to -1.

Subitems:

1 State

State of the cogging torque compensation table. Possible values:

0	NO DATA	there is no cogging table in the memory
1	IN PROGRESS	
2	SAVING	recording is done and table is being saved to the memory
3	DATA EXISTS	cogging table exists and is saved in the memory

2 Enabled

Enables or disables the cogging torque compensation feature. Set to 0 to disable or to 1 to enable the feature.

The status of the Cogging torque compensation record (0x2008:1) must be set to 1 (IN PROGRESS).

During the start-up of the device it is checked if the cogging table exists in the flash memory and if it exists, the table is communicated to a torque controller. The state is generated accordingly:

- 0 (no table exists)
- 5 (table exists)

Parameters

Object ID	Name	Description
0x2008	Cogging Torque Compensation	Parameters needed for Cogging Torque Compensation

Field Weakening

In many applications, it is required to extend the velocity range of an electric motor. In-wheel Automotive applications, gearless wind turbines and fast robotics are good examples of such applications.

As shown in Fig. 1, “Field Weakening” is a feature that increases the speed range of a PM motor by weakening the magnetic field of the rotor linearly over the speed range.

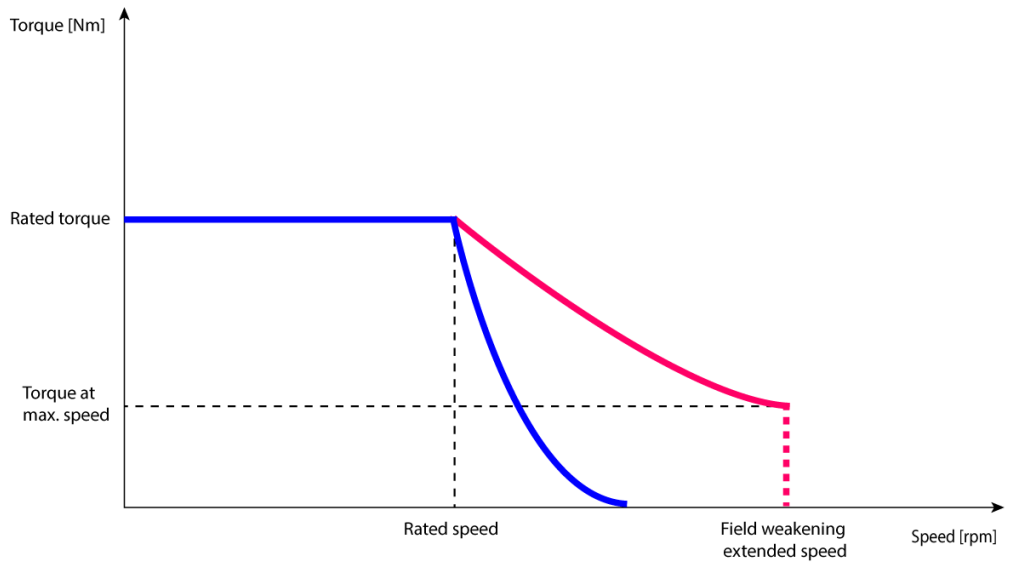


Fig. 1: Rated and extended speed range of a BLDC

Attention:

Use this feature at your own risk, it can seriously damage your hardware (including catching fire) when the values are set wrongly! Even though not all risks can be eliminate, these measures can reduce them:

- This is an expert feature! Please make sure you have sufficient knowledge about BLDC motors.
- Change all values gradually and with caution, check results often.
- Never use field weakening to achieve velocities that correspond to more than 80 Volts of peak back EMF voltage. Otherwise the drive may start burning!
- Avoid switching off the drive while in motion - especially at high velocities! Otherwise the drive may start burning!
- Avoid using the torque control mode when field weakening is activated. It is better to use velocity control or position control. Otherwise the system can become unstable, the drive may start burning or mechanical components may brake.
  Generally avoid using the torque control mode when field weakening is activated. It is better to use velocity control or position control. Otherwise the system can become unstable, the drive may start burning or mechanical components may brake.
  In some development or test scenarios, it is acceptable to still use the torque control mode, as long as there is a load machine connected to your motor or any other mechanism that will ensure that the speed cannot exceed the maximum allowable speed mentioned in (3)

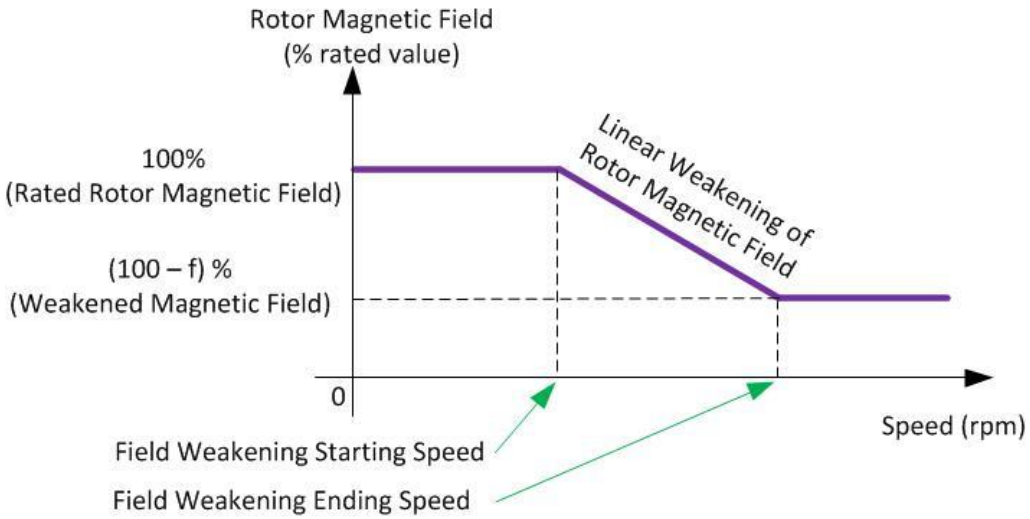


Fig. 2: Percentage of field weakening in respect to motor speed

As shown in Fig. 2, the user should set 3 parameters for this feature:

1. Field Weakening Starting Speed
2. Field Weakening Ending Speed
3. Percentage of Field Weakening (f)

Once the motor speed hits “Field Weakening Starting Speed”, the field weakening procedure starts to linearly reduce the magnetic field of rotor until the motor speed hits “Field Weakening Ending Speed”. The user can also determine how many percentage of rotor magnetic field should be finally reduced. As a result, the slope of rotor field reduction would be:

$$\frac{-f[\%]}{Field\ Weakening\ Ending\ Speed - Field\ Weakening\ Starting\ Speed}$$

> Note:
>
> The field weakening feature can not be enabled when the drive is in fault state.

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Parameters needed for Field Weakening can be found in Subindices 4-8 of the object Torque Controller

Anti-Windup Control

When the controller sends signals to the motor to accelerate at a very high rate, it's possible that the values are beyond the motor's capabilities. This is likely to happen in the case of large and long-lasting control deviations. If the controller has no knowledge about the motor's physical properties, the signal would remain saturated, even after the control error has been compensated: the motor would still accelerate at maximum power, which would result in large overshoot or even instability. Saturation limits for the integrator are introduced and can be set in the controller options (Integrator boundaries) to limit the integrator output.
The Anti-windup control algorithm is implemented with the following approach:

$$u_i(k) = u_i(k-1) + K_I \cdot T \cdot e(k)$$

$$if\ |(u_i(k))| \geq l,\ then\ u_i(k) = u_i(k) - K_I \cdot T \cdot e(k)$$

with:

e: control deviation

u_i: Output of the integral part

T: Sample time

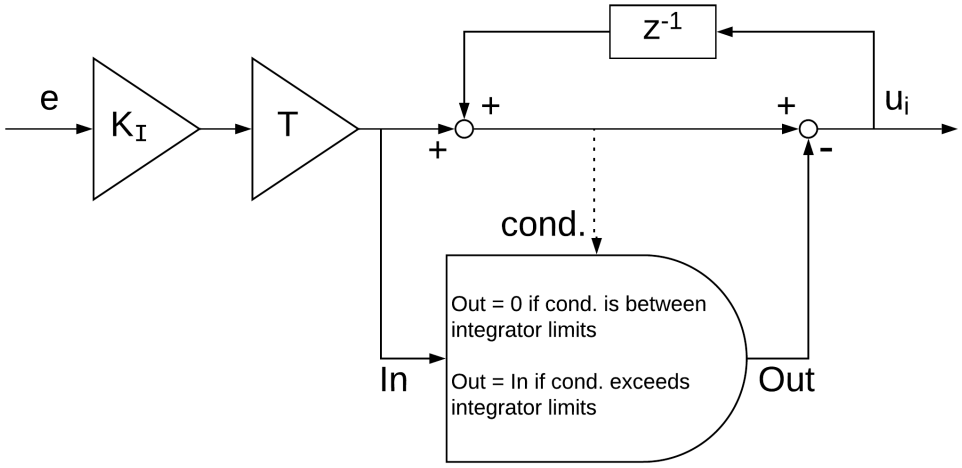
K_i: Integral gain

l: Integrator limit

Explanation: If the integral part of the controller goes higher than the value of the controller's output limit, the value that was previously added to integral is automatically subtracted.

This method is known as "clamping". It requires no tuning.

Detail view on the **integrator branch** with anti-windup:



The described implementation of the integration algorithm is used in all motion controllers (both integrators of the cascaded position controller, integrator of the PID position controller, integrator of the PID velocity controller). The variable *Integrator limit* is defined by the subitems 2012:4, 2012:8 and 2011:4 respectively.

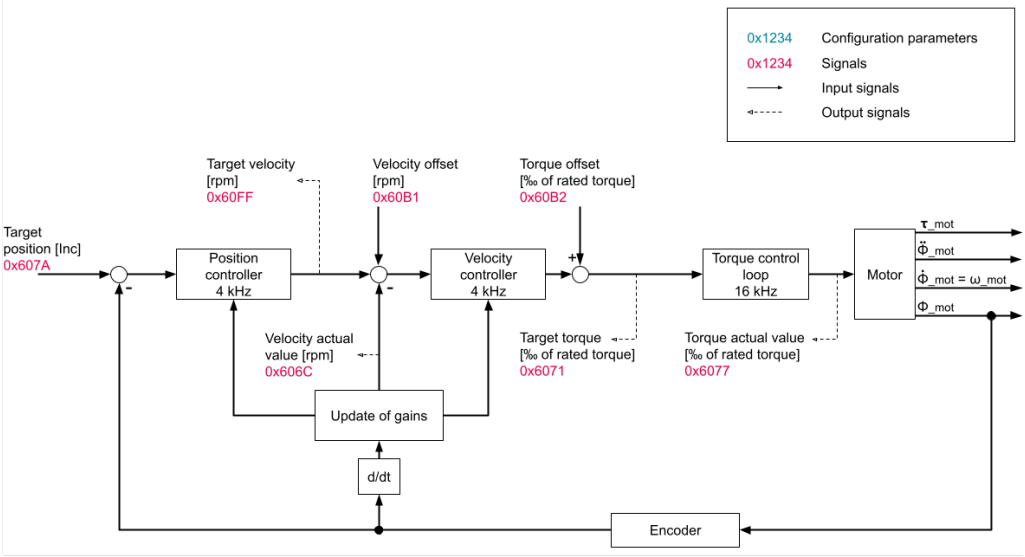
Parameters

Object ID	Name	Description
0x2011	Velocity Controller	Values K_p, K_i, K_d and Integrator Limit for Velocity Controller
0x2012	Position Controller	Values K_p, K_i, K_d and Integrator Limit for Cascaded Position Controller (Both loops: Position and Velocity)

Gain Scheduling

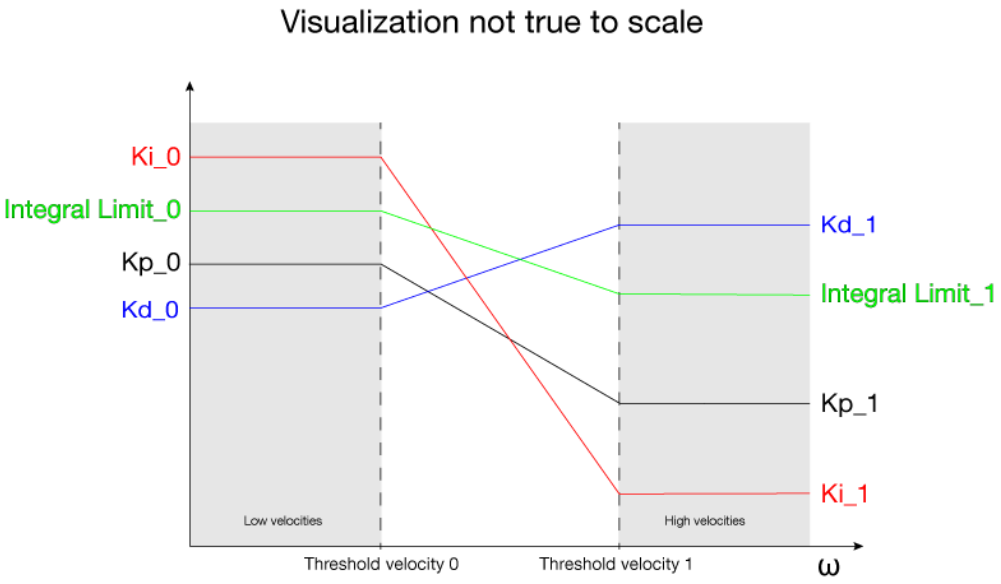
Depending on the application, the “optimal” tuning for the cascaded PID controller may differ between different operating points. If there is only one tuning available, it has to be a compromise not using the full potential of the hardware.

This feature allows the user to apply controller gains which will change dynamically regarding the current speed. This technique can be used within the cascaded position controller to compensate for nonlinear effects, especially stiction / friction in the mechanics.



For velocities (in rpm) below a user-defined value, a set of PID values determined best for slow motion is used. Likewise, for velocities (in rpm) above a user-defined value, a set of PID values determined best for fast motion is used.

The PID values between these values are **interpolated linearly.**



Note:

Please note that this feature is at the moment experimental. It is currently accessible only directly via EtherCAT through the parameter 0x2013:1 … 0x2013:18. The gain scheduling

eature is only available in Cascaded PID mode which is selected in object 0x2002.

Parameters

Object ID	Name	Description
0x2002	Position control strategy	Selects the control structure used for position control
0x2013	Gain Scheduling	Parameters needed for Gain Scheduling

Motor Overload Protection (i2t)

Overview

This feature will evaluate the “overheating” status of the motor. If the motor is overheated more than the allowed limit, this feature will limit the RMS value of phase currents to protect the motor from being overheated beyond the permitted limits.

Note:
This feature doesn’t directly measure the motor temperature. It is only a limiting feature which doesn’t allow the motor to overheat.
It is able to detect fast transients and can therefore react much faster than the attached temperature sensors.

Requirements

To use this feature, the following parameters must be set properly according to the motor data sheet:

Object ID	Name	Description
0x6075	Motor rated current	Motor rated current (rms value)
0x6073	Motor maximum current	Motor maximum current (rms value)

Object ID	Name	Description
0x200A	i2t	Parameters to activate the i2t feature

Description of Subitems for 0x200A:

Object ID	Name	Description
0x200A:1	Enable	Set to 1 to enable the protection
0x200A:2	Peak Time	Maximum time period [ms] in which the motor is allowed to have its maximum current
0x200A:3	Percent max	The percent of the maximum transient energy allowable in the motor
0x200A:4	Protection active	When this flag is set, the output current is limited to equal or less the nominal current

Operating principle

The difference between copper losses under rated conditions and overload conditions is calculated, and is used to limit phase currents in case of motor overheating.

Once the motor phase currents are higher than their rated values, the overheat calculation starts by integrating the difference between losses under rated and overload conditions.

When a current is injected that is above the nominal current, the tracking variable will rise. If the overcurrent is less then the maximal value the peak time is extended accordingly.

Once the tracking variable reaches 100%, the calculated thermal energy storage is reached and the output current in all phases will be limited to their rated values.

When the output current has fallen below the continuous limit, the difference between the actual current and the continuous limit current will be negative and the i2t tracking variable will decrease.

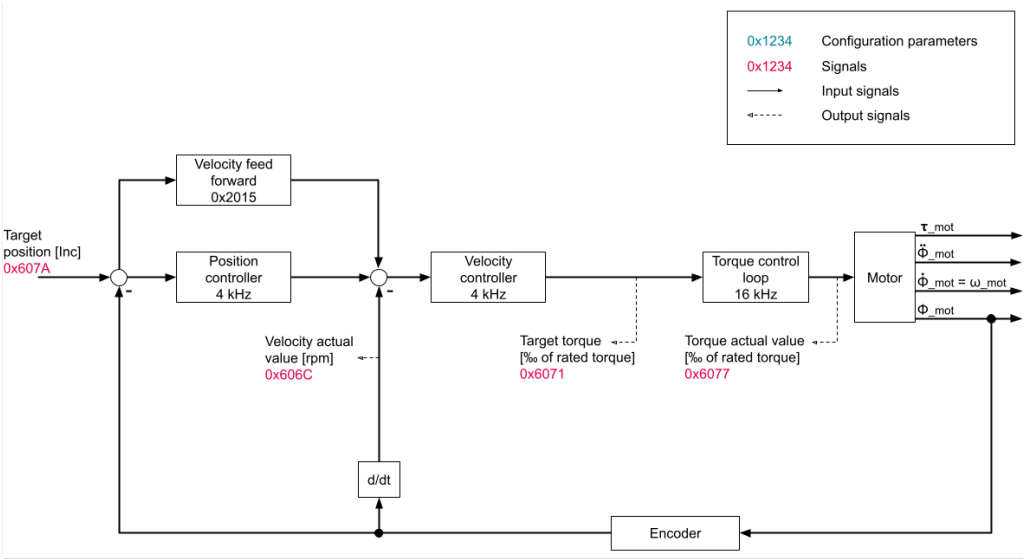
When the i2t tracking variable falls to 50%, current limiting is turned off and the motor phase currents can reach higher values again.

Attention:

The tracking variable in subitem 0x200A:3 has a maximum value of 250%. Values that exceed this limit are ignored. If the drive is used at values above 250%, the thermal energy in the motor is completely untracked!

Velocity feed forward

Velocity feed forward allows to increase performance as the target position signal gets routed directly to the velocity controller so that the velocity signal can be generated faster. After differentiation, the signal is filtered by a first-order low-pass filter and multiplied by a configured gain.



Note:

The filter is used to smooth the derivative of the target position. This is needed to soften communication lags (if present), attenuate noise in the target position signal and diminish the influence of target position quantization.

Usage

- Default value of gain is 0 (This means Velocity feed forward is off)
- The gain is given in per mil 1. A value of 1000 is a good starting point.

- Reduce the cutoff frequency if periodic torque spikes are observed during motion. The default value is 1000 Hz.
- Adjust gain and filter cut-off frequencies until torque distortions during motion are acceptable.
- To switch off the filter enter a very high cut-off frequency.

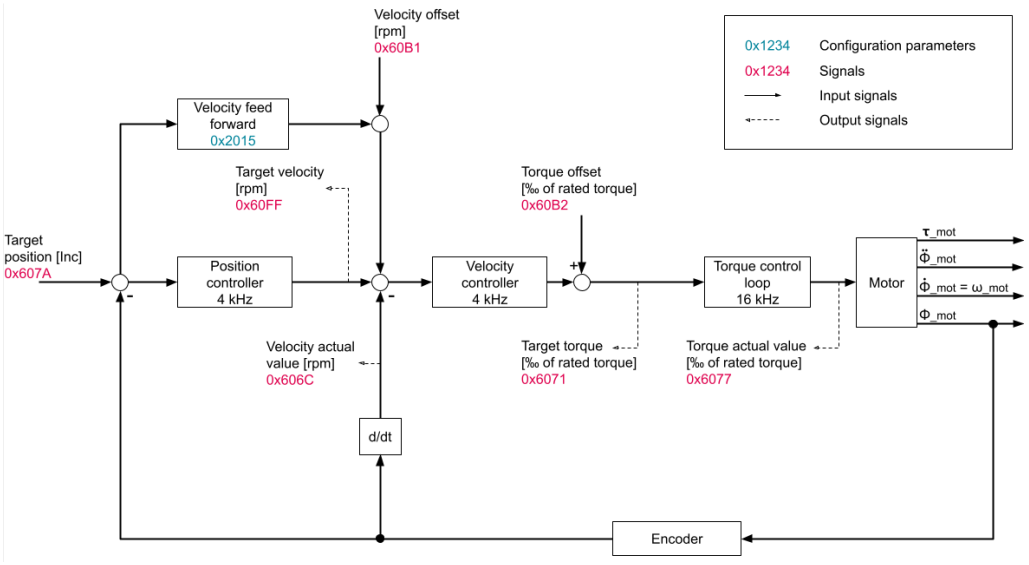
[1]

1000 means the target position signal will be routed directly, 500 means scaled to half

> Note:
> It is highly recommended to use an encoder with a resolution of more than 14 bit per revolution. Using encoders with a lower resolution will result in computed velocity quantization and torque spikes during the acceleration of the motor. Using a dual loop control might result in a similar effect if the gear reduction ratio is high and the resolution of the driving shaft encoder is small. In these cases, the feature still can be used with a very low cut-off frequency (5-10 Hz). Alternatively, a master-generated feed forward velocity can be sent to the slave via the velocity offset object.

Using a velocity offset additionally

Additionally, a velocity offset can be added to the control loop for even faster movements with high dynamic.



Click here for details for using the velocity offset.

Object used for Velocity feed forward

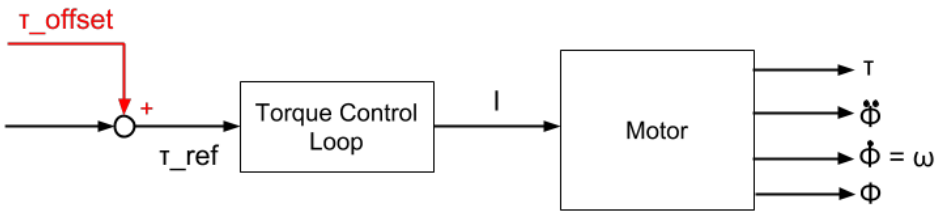
Object ID	Name	Description
0x2015	Velocity feed forward	In this object the cut-off frequency of the low-pass filter and the gain are configured

Torque Offset

This feature can be used to improve the performance of highly dynamic motion.

> Use case:
>
> Dropping a load
>
> If a load torque in a joint suddenly decreases, an involuntary movement can occur when the control algorithm tries to compensate. To avoid this peak in torque deviation, the torque-offset object can be used. The central application controller “knows” when the robot will drop the payload and how much this payload is. The robot master controller can use the kinematic model of the robot to compute how much torque this payload will add to each of the joints. So it can command these torques as offset torque to each joint. So right at the moment when the robot drops the payload, it will exert the exact amount of compensation torque that is required to stay in its desired pose.

On top of the desired velocity input, the position and velocity controllers also accept a desired ‘off-set torque’, which is a dynamically changeable torque value that will be added to the torque value generated by the velocity PI controller. This applies to the simple PID position controller, to the cascaded position controller and to the velocity controller. The offset torque can be updated at full EtherCAT communication frequency (4 kHz in most configurations) and is given in the unit of per thousands of rated torque.



To use this feature, the high level robot control must contain a model or any other knowledge about the system's physics, e.g. acceleration feed forward or gravity compensation.

The robot master controller can use the kinematic model of the robot to compute how much torque this payload will add to each of the joints. So it can command these torques as offset torque to each joint. So right at the moment when the robot drops the payload, it will exert the exact amount of compensation torque that is required to stay in its desired pose.

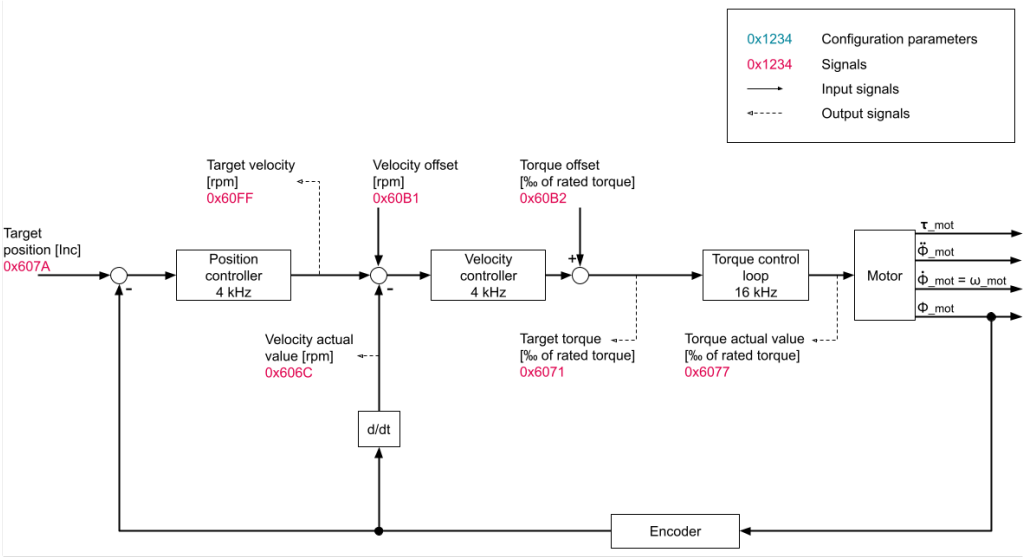
> Note:
>
> As this value should be fed dynamically with continuously updated values, it is not accessible in OBLAC Drives, but through the EtherCAT object 0x60B2.

Parameters

Object ID	Name	Description
0x60B2	Torque Offset	This object provides the offset for the torque value

Velocity Offset

To improve the positioning performance when the position reference is changing dynamically, an additional velocity offset is available in the cascaded controller.



This velocity offset must be computed by the master, it is not computed on the slave.

> Note:
>
> As this value should be fed dynamically with continuously updated values, it is not accessible in OBLAC Drives, but through the EtherCAT parameter 0x60B1.

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values K_p, K_i and K_d for Torque Controller
0x2012	Position Controller	Values K_p, K_i and K_d for Cascaded Position Controller (Both loops: Position and Velocity)
0x60B1	Velocity Offset	This object provides the offset for the velocity value

Filtering

Overview

> Attention:
>
> Since Firmware 5.0 only 1st order Position and Velocity Feedback filters are supported.

A digital filter removes unwanted components from the signal which passes through it.

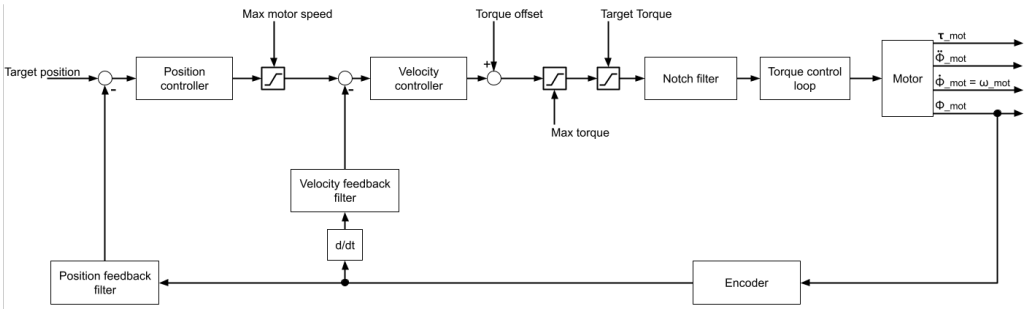
This feature can be used for example as:

- Position Feedback filter - Reduce encoder noise. A 1st order low-pass filter is supported.
- Velocity feedback filter - Reduce noise and effects of encoder nonlinearity in the velocity signal. A 1st order low-pass filter is supported.
- Loop shaping filter - Suppress excitation of resonance frequencies in the output of the motion control loop or in the input of the torque control loop. A 2nd order notch filter is supported.

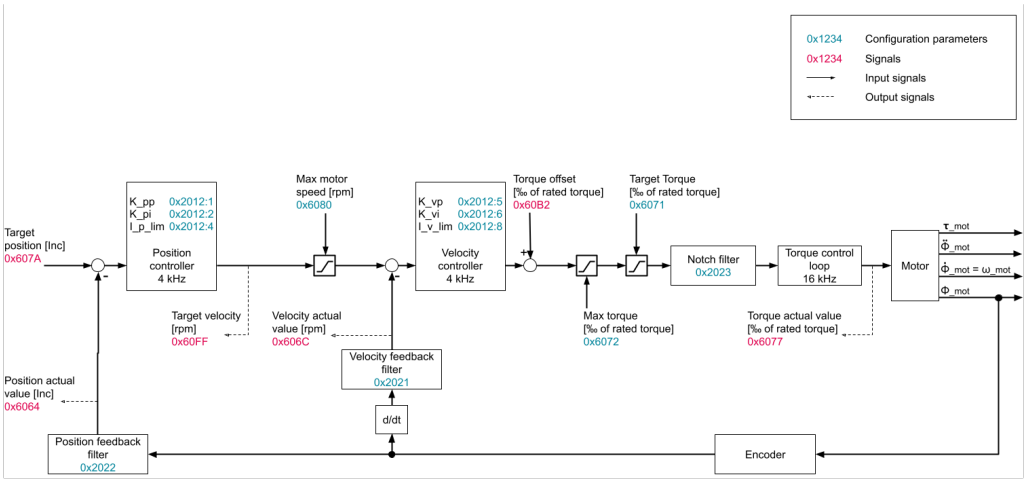
Any signal can be decomposed into a sum of sine waves with different frequencies. Some of them might negatively affect the control loops. It is possible to attenuate these unwanted frequencies by properly configuring a corresponding filter.

Configurable filters are implemented for the signals that are considered the most important for motion control.

General View



Detail View



The position feedback filter can also be used with a simple PID loop:

Methods

Definitions

- s - Laplace-domain frequency parameter
- f_c - Cutoff frequency of the low-pass filter or central frequency of the Notch filter, defined in Radians/s [1]

[1] My new footnote

- f_W - Full width of the rejection band of the Notch filter
  $\frac{f_c - f_w}{2}, \frac{f_c + f_w}{2},$

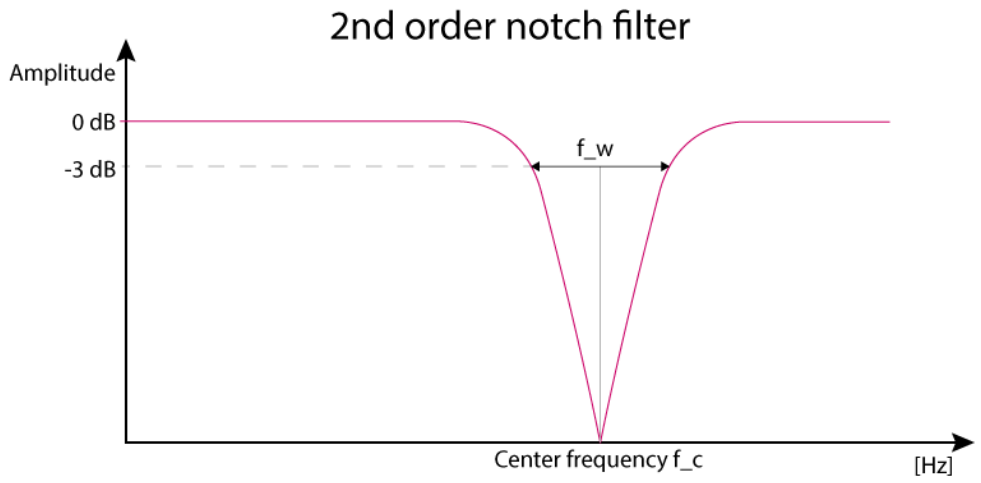
  defined in Radians/s [0]
- ζ

  - Damping ratio

  [0] Radians are used for internal calculation, user input is in [Hz]

Low-pass filter

Torque loop input shaping filter



The following equation was used to design the torque loop input shaping filter:

$$BS2(s) = \frac{s^2 + f_c^2}{s + f_w s + f_c^2}$$

Application Guide

Position and Velocity Feedback Low-pass filters

Torque loop input shaping filter

---

[0] My new footnote
[0] My new footnote

Parameters

Object ID	Name	Description
0x2021	Velocity feedback filter	Cut-off frequency of velocity feedback filter
0x2022	Position feedback filter	Cut-off frequency of position feedback filter
0x2023	Notch filter	Center frequency and bandwidth of notch filter

Position and velocity feedback low-pass filters

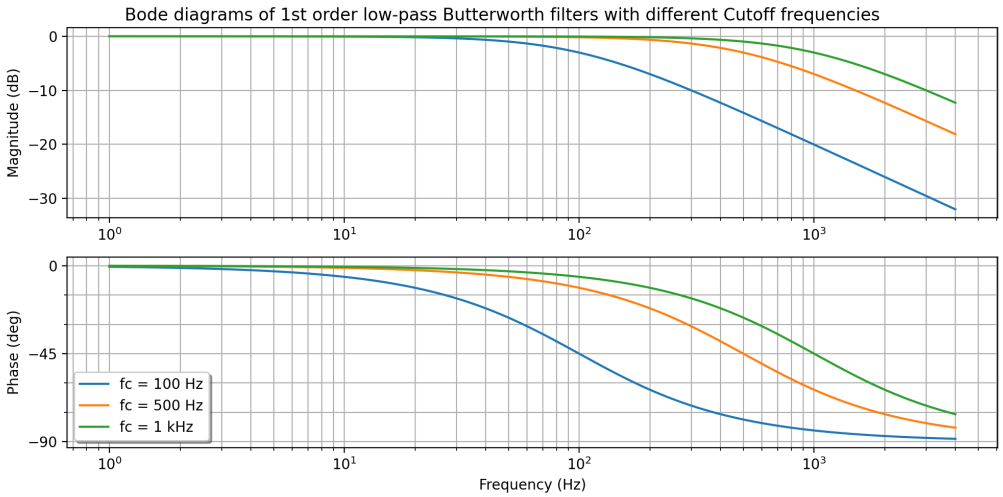
These filters are used to cope with distorted feedback signals that occur in the control loop. Typical distortions to consider are:

- Encoder noise in the position signal - optical and good magnetic encoders usually have a low amplitude. 1st order filters with a 1000 Hz [1] cutoff are a good starting point.
- Noise in the velocity signal - noise from the position signal is amplified due to differentiation. 1st order filters with a 500 Hz * cutoff are a good starting point.
- Velocity signal distortions due to encoder nonlinearity. These distortions typically appear with a frequency of 1 (dominant), 2 and 4 * motor speed [revolutions per seconds].

[1] The specified frequencies are just initial values that will not interrupt the motion control. Lower values should be found and set; see A practical guide to low-pass filter design.

> Note:
>
> Setting the velocity filter more rejective than the position filter is one possible approach, see A practical guide to low-pass filter design for details how to set up filter parameters.

The main parameter of the Low-pass filters is the cutoff frequency which determines the frequency where the filter starts to attenuate the signal to -3 dB of the amplitude. After that frequency, a roll-off region starts that further attenuates the signal.



Note:

Filtering comes at a cost of a phase shift. This might significantly limit the maximum sharpness of the control loop.

A practical guide to low-pass filter design

1. Determine the approximately required bandwidth of the control loop. This can be done based on the reference profile specifications or roughly estimated from the actual rise time during a step response test:

   $$Bw \simeq \frac{0.35}{t_r}[Hz]$$

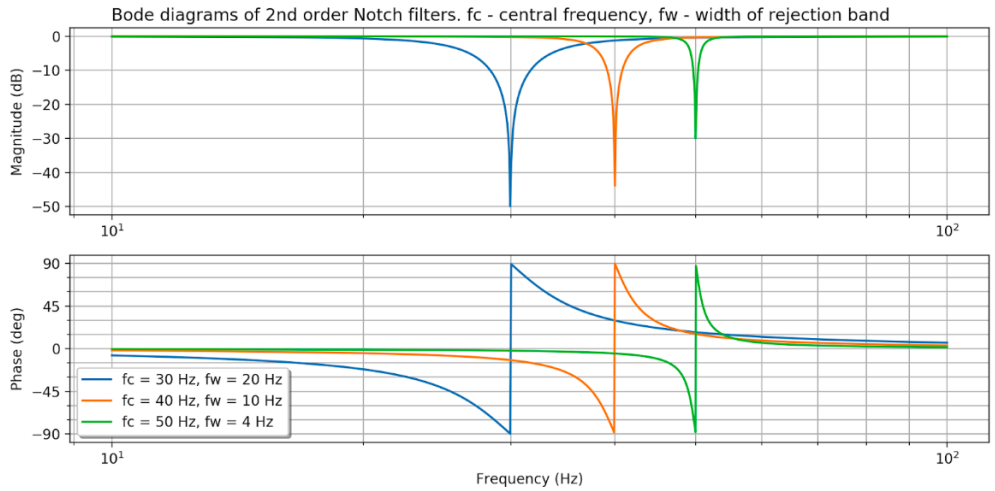
2. Ideally, the filtered signal shouldn't be modified until this frequency. The magnitude change and phase shift caused by the filter have to be as little as possible. Thus, the filter cutoff frequency should be approximately one decade higher (Bw * 10).
3. Examine the spectral density plot of the target signal and determine the frequency range that has to be filtered out.
4. Enable a filter with the default cutoff frequency.
5. Try to change the cutoff frequency until an appropriate balance between noise filtering and phase-shift/magnitude attenuation is reached.

Parameters

Object ID	Name	Description
0x2021	Velocity feedback filter	Cut-off frequency of velocity feedback filter
0x2022	Position feedback filter	Cut-off frequency of position feedback filter

Torque loop input shaping filter

If the mechanical system has a resonance (for example a natural frequency) or if the system contains another source of distortion at a known frequency, exciting this particular frequency with the output of the torque controller may result in unwanted oscillations. This filter can be used to remove a specified frequency band around the resonance.



In practice, there is a certain challenge in usage of notch filters. If the resonance frequency is too low, introducing a notch filter will impair the control signal by adding a **phase delay**. It is therefore recommended to **use notch filters only for high frequency resonances**. *

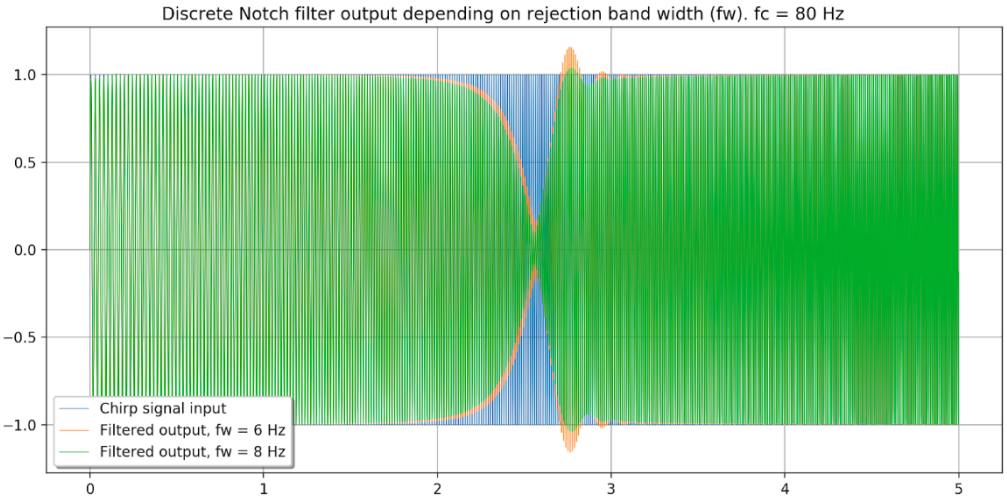
* The frequency depends on both the central frequency and the rejection width. Generally, it should be higher than 50 Hz.

A practical guide to notch filter design:

1. Find resonances of the system. This can be done by observing a frequency response during the system identification procedure: Check the velocity reaction to a specific torque input frequency.
2. Decide which resonance should be compensated and identify the width of excitation frequencies.
3. Record the reference behavior of the system before applying the notch filter.

4. Configure the filter with the obtained parameters.
5. Check the resulting performance of the setup and optimize the filter parameters.

> Note:
>
> The **smaller the width** of the rejection band, the **more ripples** are in the surrounding frequencies amplitudes. However, the area of the phase shift is smaller as is the signal attenuation at the rejection band. It is recommended to **use a rejection band width of at least 8 Hz** in order to avoid overshoot.
>
> The bigger the width of the rejection band, the smoother is amplitude transition around the central frequency. Additionally, the attenuation is bigger. This happens due to the second order of the filter



Parameters

Object ID	Name	Description
0x2023	Notch filter	Center frequency and bandwidth of notch filter

Command smoothing and Interpolation

Overview

Commands are sent to the slaves with a step size that depends on the master's update rate. In EtherCAT, these steps are usually in intervals of 250 µs to 8 ms while other protocols may have

even slower rates.

These steps may cause audible noise or even instability when a controller is tightly tuned.

Fast command changes and slow update rates increase the negative effects on control performance.

To improve the quality of the trajectory, the input command can be shaped with a finite impulse response filter.

This filter smoothes the control values from the previous to the current target, smoothing out the input and preventing spikes in the derivative around the target changes.

This feature can be used in all position and velocity control modes:

- Cyclic synchronous position mode
- Cyclic synchronous velocity mode
- Profile position mode
- Profile velocity mode
- Homing mode

Side Effects

> Attention:
> This feature introduces delay to the control input therefore it's important to close the control loop in the servo drive directly. If the control is closed on the master, this feature would introduce a lag that may affect that controller's stiffness.

> Note:
> All axes on the chain must be commanded together and should have the same FIR settings.

Applications

Input interpolation

The master is configured to use the distributed clock so that all input commands are equally spaced in time. The FIR filter is set to operate such that the filter order is equal to the number of control cycles between two successive target command updates. This ensures the command is fully respected, since the last step of interpolation hits the requested point.

Pros

- The controller “ramps up” to the next commanded position instead of seeing a step function.
- The controller can be tuned much tighter than it would normally
- Removes noise that can occur when a tightly controlled system receives a step input.

Cons

- See Side Effects

Jerk limiting

When the input trajectory contains undesirable discontinuous or aggressive motion, the performance of the mechanical system can be unsatisfactory.

Fixing the input trajectory can be difficult if the trajectory is built from sensor input and dynamically updated. Some trajectory generators are very basic, and smoothing the input is ideal to avoid poor motion.

The FIR filter should be set to smooth the input to the desired magnitude. Higher filter order will result in a smoother trajectory. Lower filter order will result in closer following of the input command.

Pros

- An easy way to avoid the need for complex trajectory generators on the master.
- Stable smoothing that guarantees settling on the input command by the end of the filter order.

Cons

- Smoothing will prevent the actuator from following the commanded trajectory, which may be problematic when part of a trajectory needs to be followed closely while other parts should be smoothed.
- End effector precision may be poorly affected for robots with serial kinematics.

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

> Note:
>
> The disadvantages above can be mitigated by slowing the trajectory in areas that need more precision.

Communication jitter mitigation

When no distributed clock is available, the clocks on the slave and master devices will drift, and this will cause a cycle to periodically either miss a command, or complete before the command arrives to the controller. This will effectively either double the input command or see no change to the input command. This creates a discontinuity that is twice the magnitude of the normal command steps.

Set the filter to an order similar to what should be chosen for input interpolation.

Pros

- While distributed clock and interpolated input will yield the best performance, using the FIR filter here will also smooth the discontinuities caused by communication jitter.
- The filter order can be boosted for additional smoothing if the communication delays are severe.

Cons

- See Side Effects
- The trajectory is not guaranteed to match the input, since large steps will still cause “bumps” in the command. These bumps will be much less aggressive, but they won’t be entirely removed from the input.
- Extra smoothing will always yield extra deviation from the original command.

Detailed Description

A boxcar filter with rectangular gains is implemented, equivalent to a moving average filter (MAF). A moving average filter is equal to a linear interpolation if the filter order is equal to ratio between master and slave sample time - 1.

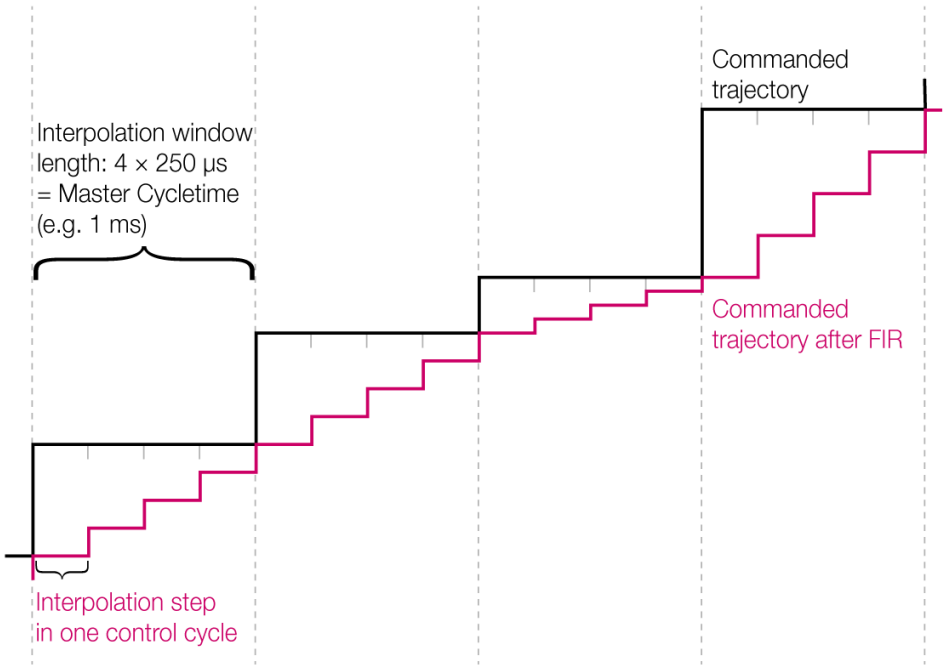
$$Filter\ order = \frac{master\ cycle\ time}{slave\ sample\ time} - 1$$

An example of the filter order for a master with 1 ms cycle time and 250 µs cycle time SOMANET slave:

$$3 = \frac{1ms}{250\mu s} - 1$$

The filter order is defined in multiples of 250 µs (e.g. 3 for a master with a 1 ms cycle), 3 is the default value that suits applications with 1 kHz EtherCAT masters perfectly. This number defines the length of the ring buffer that will be used as a MAF storage. At each 4 kHz cycle, the values in the buffer are summed up and divided by the buffer length to produce the output.

Command smoothing



Usage

In object 0x2027, the expected update rate and/or desired smoothing can be configured:

- Subitem 1 switches the feature on (1) or off (0)
- Subitem 2 selects the order of the filter

The Order is allowed between 0 (no action) and 31. The filter is enabled by default with an order of 3.

The order reflects the amount of steps the algorithm “looks back”.

Parameters

Object ID	Name	Description
0x2027	Control input FIR filter	Parameters needed for Finite impulse response Filter

Control supervision

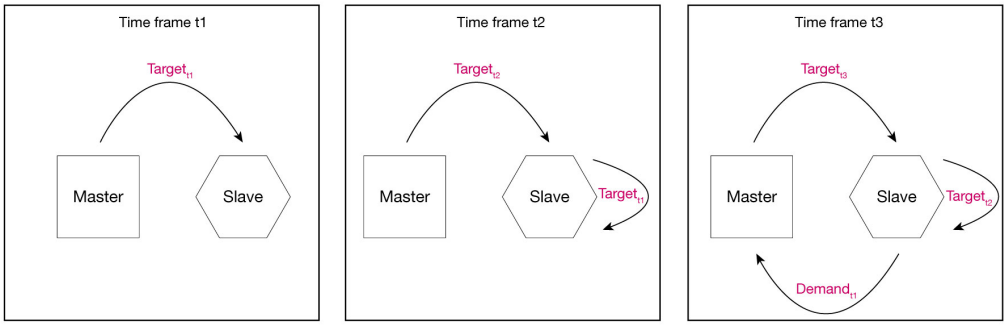
This feature allows to supervise the work of the control loops. This includes monitoring the internal variables and basic analysis of the control loop performance. Some monitoring functions and data objects are available only in specific control modes.

The main supervision functions are:

- Following error - Basic function that needs to be calculated on the master
- Following error window - Advanced feature that raises a flag in the Statusword
- Target reached function - Raises a flag in the Statusword

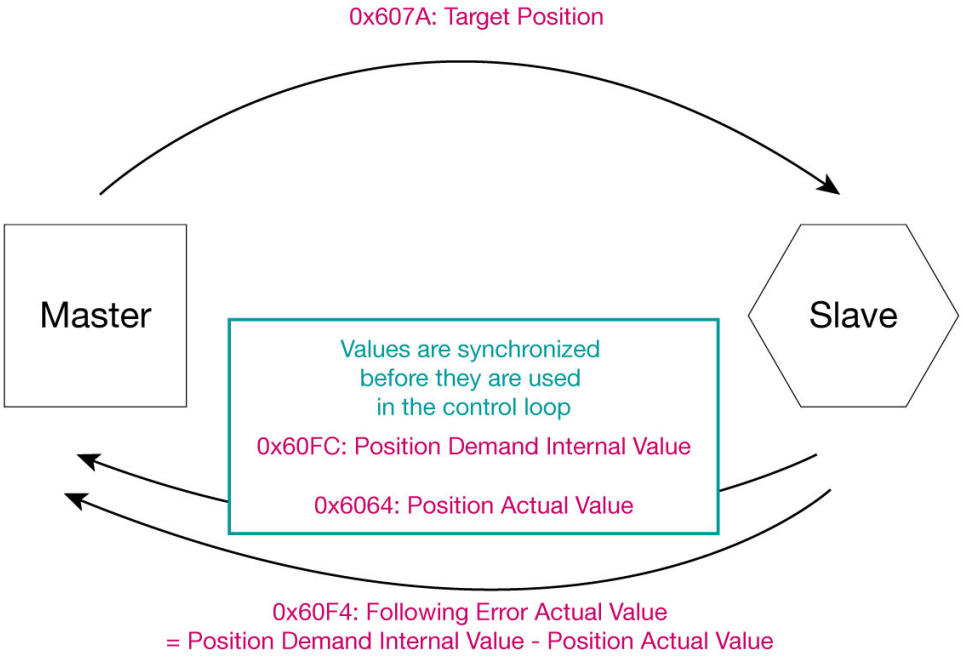
Timing diagram of the supervision objects

Two communication cycles after the master sends a target value to the slave (position/velocity/torque), the slave sends back two values: the Demand Value, indicating the value that was received by the slave and the Actual value as read from the sensors. These values are synchronized and sampled from inside the control loop.

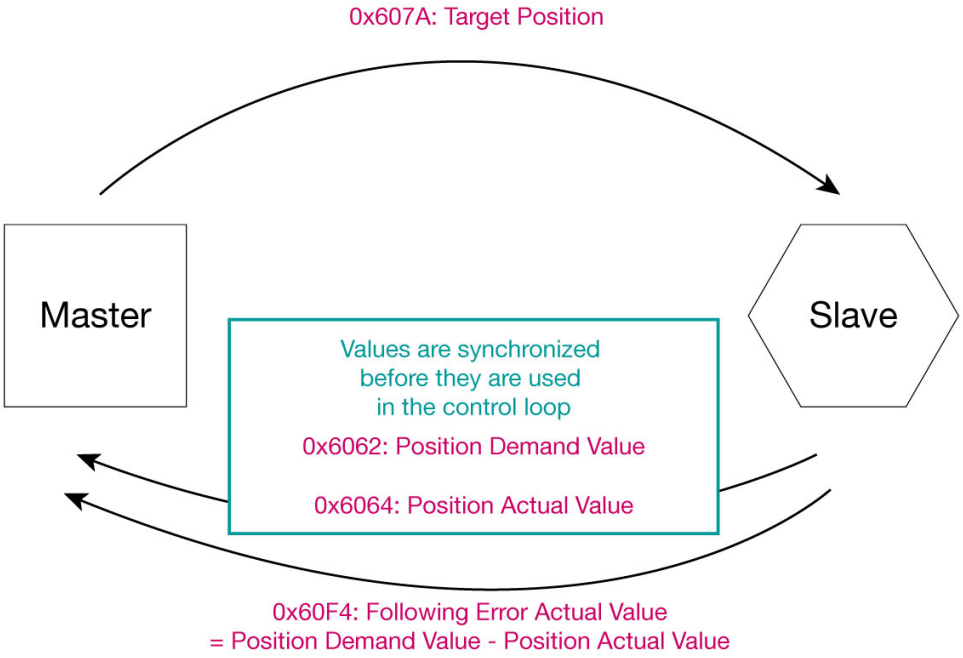


Name and designation of these objects differ, depending on the chosen mode of operation:

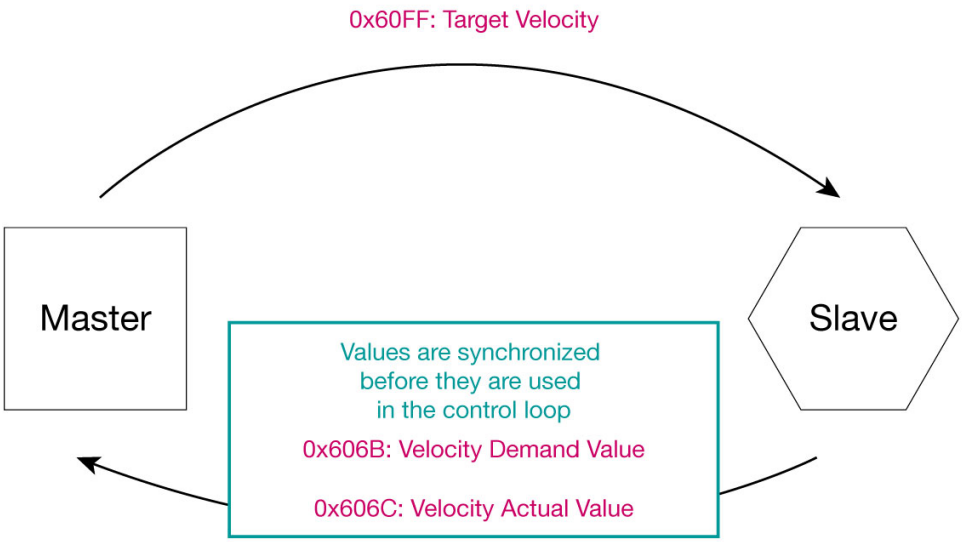
Profile Position


0x607A: Target Position
Master
Slave
Values are synchronized
before they are used
in the control loop
0x60FC: Position Demand Internal Value
0x6064: Position Actual Value
0x60F4: Following Error Actual Value
= Position Demand Internal Value - Position Actual Value

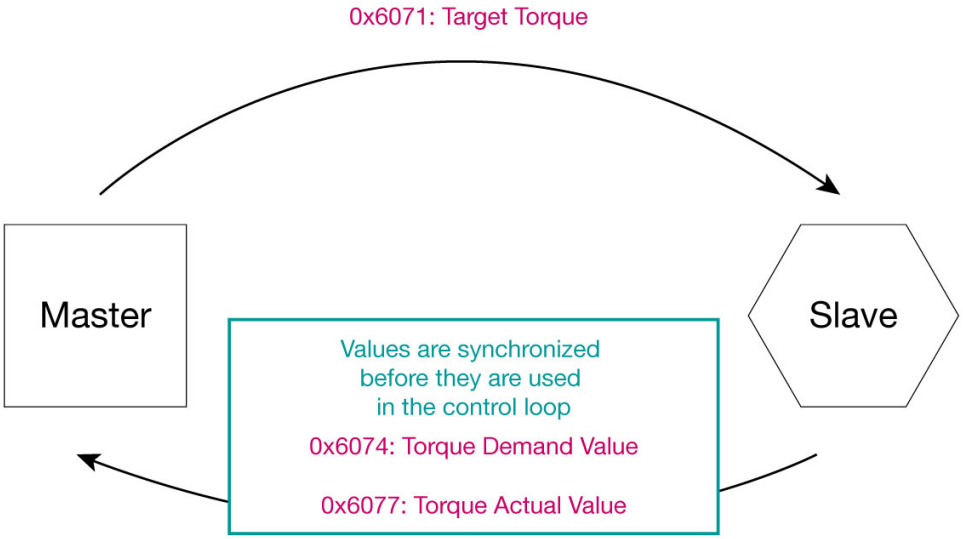
CSP


0x607A: Target Position
Master
Slave
Values are synchronized
before they are used
in the control loop
0x6062: Position Demand Value
0x6064: Position Actual Value
0x60F4: Following Error Actual Value
= Position Demand Value - Position Actual Value

CSV



CST



Note:

In Profile Position mode, the master does not send a new calculated position at each cycle,

instead, the slave computes the intermediate positions itself directly at the trajectory generator.

Therefore, the objects sent back from the slave to the master are different than in CSP.

Following error

The easiest way to measure the performance of a control loop is to calculate a following error:

Following error = Target value - Actual value

If the master-generated value is used as a target value, it is not synchronized with the actual measured value. The typical delay can be up to 3 communication cycles. Thus, the computed error will contain an extra error due to communication delay.

To circumvent this issue, 'Demanded value' objects should be used. These values a sampled synchronously with the actual measured value and should be used to correctly compute the following error.

Example: Velocity following error = Velocity demanded value - Velocity actual value

Alternatively, the object Following error actual value (0x60F4) can be used to monitor the position control loop. This object is also used for the Following Error Window feature.

Following error window

If the actual value is outside of the configured window for a configured period of time, the user is informed that the following error is bigger than the configured value.

This information is reflected by bit 10 of the statusword.

Different options are available for the subsequent action.

Use case:

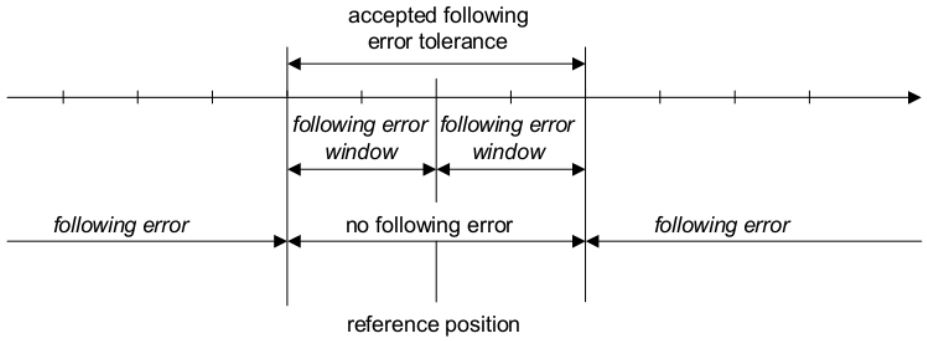
Predictive maintenancing (monitor system wear)

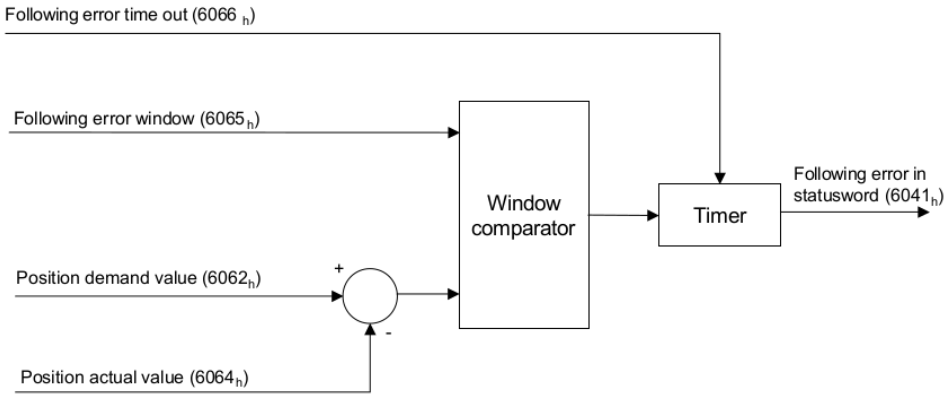
In case the following error grows bigger in time, check if mechanical wear is the reason.

Note:

The Following error window feature can only be used in a position control mode (operational modes CSP and PP)

Definitions





Configuring the Following error window

- Following error window 0x6065 [inc]: Use this object to indicate half of the configured range of tolerated position values symmetrically to the position demand value. If the position actual value is out of the following error window, a following error occurs. If the value of the following error window is FFFF FFFF h , the following control is switched off (default).
- Following error timeout 0x6066 [ms]: Use this object to indicate the minimum time that the actual position must be out of following error window so that the bit 13 of the statusword is raised to 1, indicating that a Following error has occurred.

Result

- Bit 13 of the Statusword indicates that the following error is outside the allowed window:
  - 0: value within error tolerance
  - 1: value exceeds error tolerance
- Following error actual value 0x60F4 [inc]: Provides the actual value of the following error.

Following error option codes

These codes are to configure the drives reaction in case of a following error that exceeds the window.

Value	Definition
0	Drive will report Following error in statusword (bit 13) and stay in operational state
1	Drive will report Following error in statusword (bit 13), perform a quick stop and enter a fault state
2	Drive will report Following error in statusword (bit 13) and immediately enter a fault state

Target reached function

This feature can be used to inform the user if the requested target was reached (for example if the target position was set in the Profile mode). A specified target (position, velocity or torque) can be defined with a margin at each side.

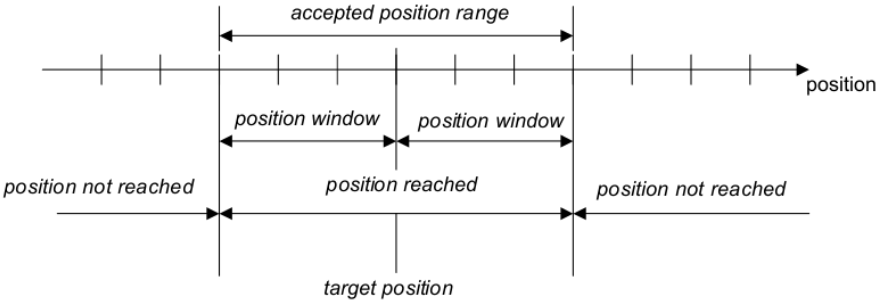
If the actual value is inside of the accepted position range for a configured period of time, the user is informed that the target position has been reached.

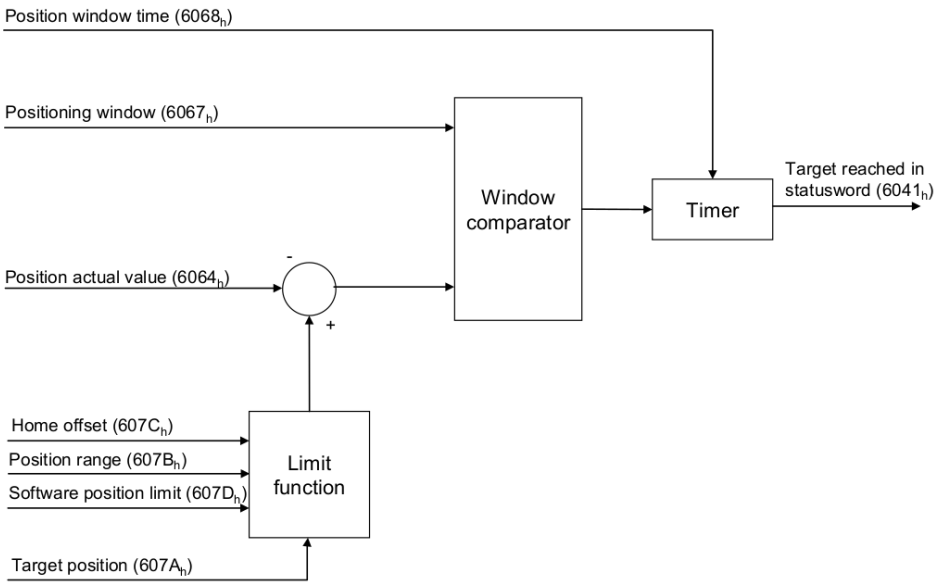
This information is reflected by bit 10 of the statusword.

> Use case:
>
> Pick and place operation
>
> In case the real position where the good is to be picked up or placed has more margin than

the precision of the motion control system, using a wider position window allows a faster operation.

Definitions





Configuring the position window (CSP)

Note:

The accepted position range is calculated as two times the position window.

- Enter the position window value in object 0x6067 Position window (unit is given in increments of the position encoder). The value 0xFFFFFFFF switches the feature off (default).

- In object 0x6068 Position window time you can configure the time required for the position sensor to remain within the position window before it's considered to be inside the window. The unit is given in ms.

Configuring the velocity window (CSV)

> Note:
>
> The accepted velocity range is calculated as two times the velocity window.

- Enter the velocity window value in object 0x606D Velocity window (unit is given in rpm) The value 0xFFFF switches the feature off (default).
- In object 0x606E Velocity window time you can configure the time required for the value to remain within the velocity window before it's considered to be inside the window. The unit is given in ms.

Configuring the torque window (CST)

> Note:
>
> The accepted torque range is calculated as two times the torque window.

- Enter the torque window value in object 0x2014 Torque window subitem 1.
- In subitem 2 you can configure the time required for the torque value to remain within the torque window before it's considered to be inside the window. The unit is given in ms.

Result

Bit 10 of the Statusword indicates that the position is within the allowed window:

- 0: value outside accepted position range (position not reached)
- 1: position reached

Monitoring the internal variables

To examine the inputs and outputs of a given controller, the following objects can be used:

- Position demand value 0x6062: Provides the value of the position sent to the position controller after position limits are checked. (See 0x607D Software position limit.) The units are in increments of the position encoder.

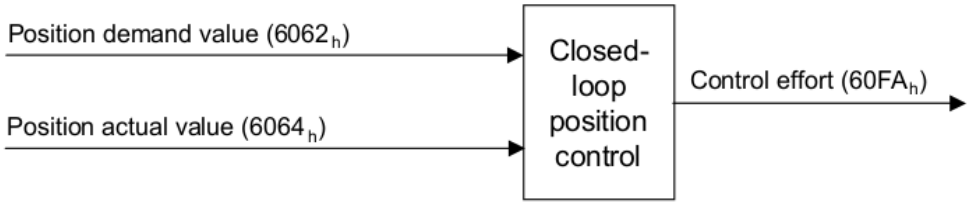
- Position demand internal value 0x60FC: Provides the output of the trajectory generator in profile position mode. The units are in increments of the position encoder.
- Position value 0x6064: Provides the actual value of the position encoder. The units are in increments of the position encoder.
- Velocity demand value 0x606B: Provides the output value of the trajectory generator in velocity control, and the output of the position controller when using the cascaded controller. The units are user-defined (RPM by default; see 0x60A9 SI unit velocity to change it).
- Velocity value 0x606C: Provides the actual velocity value derived either from the velocity sensor or the position sensor. The units are user-defined (default is RPM; see 0x60A9 SI unit velocity to change it).
- Torque demand 0x6074: Provides the output value of the torque trajectory generator. The units are per thousand of rated torque.
- Torque value 0x6077: Provides the actual value of the torque, and is computed based on the currents measured in the motor. The units are per thousand of rated torque.

Control Effort

The control effort (object 0x60FA) is the output of the controller for the selected mode.

It provides a measure of the control effort (the output of the position control loop). The units are in user-defined velocity units (RPM by default; see 0x60A9 SI unit velocity to change it).

- If cascaded position control is used, the control effort is velocity demand value
- If velocity control is used, the control effort is torque demand value



Parameters

Object ID	Name	Description
0x2014	Torque window	Indicates the accepted torque values and time

Object ID	Name	Description
0x2017	Following error option codes	Configures drive's response in case of Position following error event
0x60F4	Following error actual value	Provides the actual value of the following error
0x6065	Following error window	Half of the configured range of tolerated position values
0x6066	Following error timeout	Minimum time that the actual position must be out of the window so that a following error is triggered
0x6067	Position window	Half of the accepted position range
0x6068	Position window time	Time required for the position sensor to remain within the position window
0x606D	Velocity window	Half of the accepted velocity range
0x606E	Velocity window time	Time required for the value to remain within the velocity window
0x6062	Position demand value	Value of the position sent to the position controller after position limits are checked
0x60FC	Position demand internal value	Provides the output of the trajectory generator in profile position mode
0x6064	Position value	Provides the actual value of the position encoder

Object ID	Name	Description
0x606B	Velocity demand value	Value of the trajectory generator in velocity control / output of the position controller in cascaded control
0x606C	Velocity value	Provides the actual velocity value derived either from the velocity sensor or the position sensor
0x6074	Torque demand	Provides the output value of the torque trajectory generator
0x6077	Torque value	Provides the actual value of the torque
0x60FA	Control effort	Output of the controller for the selected mode

Distributed clocks (DC clocks)

Overview

Synchronization via distributed clocks allows to share the same system time between the master and all slaves.

This feature allows to achieve high performance, especially with sharp tuning.

Without synchronization, unwanted vibration can occur when a slave's internal clock isn't synchronized with the master time. Especially when several slaves try to perform complex motion synchronously, periodical vibrations, audible noise and spikes in the torque signal can appear and the slaves may not execute the motion fully synchronously.

This is most prevalent in cyclic modes, since in profile modes the slaves compute their trajectories themselves.

> Important:
>
> For good performance, always enable the feature if possible. Ensure that your master supports it!

Distributed Clock configuration

SOMANET supports 2 DC modes:

- Slave mode - Master device will be used as a reference clock. All the slaves in the chain will be adjusted according to the master time.
- Master mode - One EtherCAT device will be used as a reference clock. Typically, the reference clock is the first slave with DC capability between master and all the slaves to be synchronized (DC slaves).

This feature has to be configured on the master side. We recommend user to use the Master mode as it has better performance than the Slave mode.

Additionally, the synchronization options have to be configured:

- SYNC 0 has to be enabled with recommended cycle time equal to the EtherCAT communication.
- SYNC 0 Shift time is usually 0.
- SYNC 1 is not supported yet and needs to be disabled.

Parameters

Object ID	Name	Description
0x20F3	DC synchronization	The status of the selected network synchronization mechanism

OS command

Overview

The OS command is a standard way of triggering actions or services on the drive by which cannot be adapted to a single SDO upload/download.

The following commands are currently available:

Command 0 Encoder register communication

Command 1 iC-MU calibration mode

Command 2 Open Loop Field Mode (OLFM)

Command 3 HRD streaming

Command 4 Motor phase order detection

Command 5 Commutation offset detection

Command 6 Open phase detection

Command 7 Pole pair detection

Command 8 Phase resistance measurement

Command 9 Phase inductance measurement

Command 10 Torque constant measurement

Command 13 Skipped cycles counter

Command 14 Ignore BISS status bits

Usage

The OS command feature can be operated with the objects 0x1023 (OS command) and 0x1024 (OS command mode).

0x1023: OS command

Select the command that should be used and start it by writing the desired value in subindex 1.

The status can be read in subindex 2.

The response can be fetched in subindex 3, it includes the status as written in subindex 2.

> Note:
>
> To calculate the progress, please subtract 100 from the value of 0x1023:2.

Subindex	Description	Data Type	Value
1	Command	OCTET_STRING	**Byte 0**: OS command ID **Byte 1-7**: Service request data A write access to the subindex will execute the command

Subindex	Description	Data Type	Value
2	Status	UNSIGNED8	**0**: last command completed, no errors, no response data **1**: last command completed, no errors, response data available **2**: last command completed, error, no response data **3**: last command completed, error, response data available **100-200**: command executing with percentage (100 = 0%, 200 = 100%) **255**: command executing (if percentage display is not supported)
3	Response	OCTET_STRING	**Byte 0**: identical with subindex 2 **Byte 1**: unused **Byte 2-7**: Service response data

Details

- Subindex 1 can't be written until the command is finished and the response is read in subindex 3. If a command is written in that case anyway, it will be ignored and an OsCmdCol error will be raised.
- The status of the command can be read in Subindex 2 and is also available in the first byte of subindex 3.

> Note:
> It is recommended to read the status from the first byte of subindex 3 instead of subindex 2 because reading subindex 2 when a command is finished will not make subindex 1 writable again. The subindex 3 has to be read at least once after writing, before the next write command can be performed.

- Both subindex 1 and subindex 3 are octet strings of size 8.
- Byte 0 of the subindex 1 is used to choose the command that shall be executed.
- The rest of the bytes are used to specify the parameters of the command, so they are command specific.
- Byte 0 of the subindex 3 shows the status of the command. Depending on its value the rest of bytes on that subindex differ:
    - If the command is in progress (byte 0 is 100-200 or 255) or completed with no response data (byte 0 is 0 or 2), no other bytes will be set.
    - If the command is completed with no errors and with response data (byte 0 is 1), command specific bytes will be set starting at byte 2.
    - If the command is completed with errors and with response data (byte 0 is 3), byte 2 will be the OS error code (see below) and command specific bytes will be set starting at byte 3.

The following table shows all the valid responses (subindex 3) that any OS command might respond with. Grey table cells mean: these bytes are optional.

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (with per-	100 - 200							

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
centage)								
Command in progress (without percentage)	255							
Command completed without errors and without response	0							
Command completed without errors and with response	1		Data	Data	Data	Data	Data	Data
Command	2							

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
completed with error and without response								
Command completed with error and response	3		OS error code (see below)	Data	Data	Data	Data	Data

OS error code

Indicates the reason for an error when the current command is completed with errors and with response data (status is 3).

> Note:
> Error codes 0 - 8 are command specific, while error codes 251 - 254 are general error codes.

The command-specific OS error codes will be explained in detail in the corresponding section.

General OS error codes:

- Unsupported command: the command ID received doesn't exist.
- Command timeout: A request to execute an OS command or to abort it was not acknowledged by any downstream service after 5s.

- Command aborted: The abort action triggered by the user was successful.
- Command not allowed: Certain commands can execute only under certain conditions (in a specific operation mode for example). This error code will be used to signal that these conditions are not met.

	OS Command #										
	0	1	2	3	4	5	6	7	8	9	1-0
OS error code 0	Invalid response				Initialization error			Open terminal A	Current ampl. error		
OS error code 1	Register not allowed				Streaming error			Open terminal B			
OS error code 2					Duration value			Open terminal C			
OS error co-					Data index value			Open FET A			

	OS Command #										
	0	1	2	3	4	5	6	7	8	9	1-0
de 3								high			
OS error code 4					Action value			Open FET A low			
OS error code 5								Open FET B high			
OS error code 6								Open FET B low			
OS error code 7								Open FET C high			

	OS Command #										
	0	1	2	3	4	5	6	7	8	9	1-0
OS error code 8								Open FET C low			
...		(reserved for future use)									
OS error code 25-1		Command not allowed									
OS error code 25-2		Command aborted									
OS error code		Command timeout									

	OS Command #									
	0	1	2	3	4	5	6	7	8	9	10
25-3											
OS error code 25-4		Unsupported command									

0x1024: OS command mode

This object is only used to abort the current command when value 0x03 is written. Otherwise (value 0x00 is written as a default) the next command is executed immediately.

Value	Description
0x00	Execute the next command immediately (default)
0x03	Abort the current command and all commands in the buffer

Details

- This object is write-only.
- Writing 0 to this object will enable “Execute the next command immediately”, which is the default mode. In this mode, new OS commands will be executed as described above.
- Writing 3 to this object will abort the currently executing command. Once the command is aborted successfully, the status “last command completed, error, response data available” (3) will be sent by the slave, and the OS error code (which will be explained in more detail later) in byte 2 of the response will signal the command abortion. If there is no command in progress, this write operation will be ignored.

OS command behavior

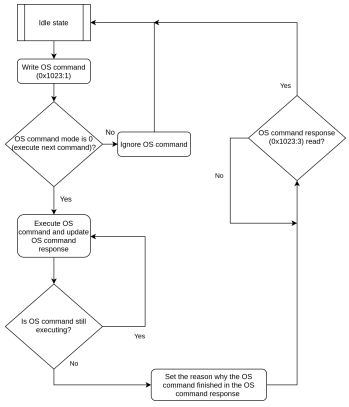
The drive only accepts OS commands when the OS command state is idle, which means that the last command was completed and the response was read from 0x1023:3. Otherwise the OS command will be ignored, and the drive will enter the Fault CiA402 state (error report description will be OsCmdCol, which stands for OS command collision).

If subindex 0x1024:0 is not 0, the OS command will be ignored and discarded and no error will be raised.

Whenever an OS command is sent, the status will change from 0, 1, 2 or 3 to 255 immediately to ensure that the correct response is received.

At any point while a command is in progress, that command can be aborted writing 3 to the object 0x1024.

Flowchart of the Command



Available commands

Command 0: Encoder register communication

This command is used for reading or writing registers from a BiSS encoder.

> Note:
>
> For more information about how BiSS register communication works, please refer to the official BiSS-C Interface Protocol Description.

Command request

The values in 0x1023:1 for this command are as follows:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0							
1						Encoder ordinal		
2	Slave address / ID (0)							¬R/W
3	Register address							
4	Register write value							

Explanation of the parameters:

- Encoder ordinal: select the target encoder for the command.
  A value of 1 will select the Encoder 1 (which is the encoder configured in 0x2110), a value of 2 will select Encoder 2 (which is the encoder configured in 0x2112).

Value	Encoder
1	Encoder 1
2	Encoder 2

- Slave address: selects the address of the slave to read (BiSS protocol level). This value must always be 0.
- ¬R/W: Selects the direction of the command: 0 for reading a register, 1 for writing it.
- Register address: Register to be read/written.
- Register write value: Value to write to the register. This is only used if the parameter ¬R/W is set to 1.

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in	255							

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
progress (without percentage)								
Command completed without errors and with response	1		Register value					
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The register value is the value read/ written from/to the selected register.
- **Status 3**: The command has finished unsuccessfully. The error code will be sent, either as a general OS error code or a command-specific OS error code:

OS error code	Error name	Error description
0	Invalid response	There was an error while executing the BiSS register transaction. This can be caused by • a too short BiSS timeout, • too high BiSS clock frequency or • trying to access a register that does not exist
1	Register not allowed	The register can't be read given the current configuration. Currently the only registers that are not allowed to read are the iC-MU I2C registers (range 0x5B-0x6F) in Internal encoder 1 of Circulo if multiturn is disabled. If this happens, it can be avoided by setting the subindex "Status bits active value" to "Status bits ignored".

Example

To write the value 0x08 in register 0x11 of the encoder configured as Encoder 2, the bytes in 0x1023:1 should be set like this:

Byte	Value
0	0
1	2
2	1
3	0x11
4	0x08

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0811010200 --type int64
```

> Note:
>
> This command can only be used with IgH ethercat Master for Linux

To fetch the response, the following command should be entered:

```
ethercat upload 0x1023 3 --type int64
```

This value should be returned: 0x0000000000080001 (Status 1, with a confirmation of the written register value).

Command 1: iC-MU calibration mode

This command is used to be able to set a particular BiSS mode when using an iC-MU encoder, which is mostly used for iC-MU calibration. The following modes are supported:

- Configuration mode: the BiSS service reads the encoder normally but it only uses the bits for register communication and discards the rest, so the encoder data is not updated. This also means *BisCrc* error is not raised while in this mode.
- Raw mode: The BiSS service reads the encoder assuming it was correctly configured for raw mode. It continues calculating all the encoder data but in addition to that it also average of the raw data and makes it available in object 0x2704 User MISO like this:

Bit		
31-28	27-14	16-0
0	Nonius count	Master count

- Standard mode: The BiSS service clocks the encoder normally.

When transitioning from any mode to Configuration mode, the current position is saved. When switching to Raw mode, this saved position will be used as a starting point for the position calculation, because the raw data is relative. This means that the motor should not be moved while in Configuration mode if we plan to switch to Raw mode after.

Command request

Values in 0x1023:1 for this command:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	1							
1				Mode enable		Encoder ordinal		

Explanation of the parameters:

- Mode enable: select the iC-MU calibration mode.

Value	Mode
0	Configuration mode
1	Raw mode
2	Standard mode

- Encoder ordinal: select the target encoder for the command.

  A value of 1 will select the Encoder 1 (which is the encoder configured in 0x2110), a value of 2 will select Encoder 2 (which is the encoder configured in 0x2112).

Value	Encoder
1	Encoder 1
2	Encoder 2

Command response

Possible responses of this command:

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without per-	255							

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
centage)								
Command completed without errors and without response	0							
Command completed with error and without response	2							

- **Status 255**: The command is in progress.
- **Status 0**: The command was finished successfully.
- **Status 2**: The command couldn't be executed on the sensor. The received Mode enable value is unknown.
- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

Configuring raw mode on encoder configured as Encoder 1:

1. Enable Configuration mode to freeze the current absolute position.

   To change Internal encoder 1 of Circulo to Configuration mode, the bytes in 0x1023:1 need to be set like this:

Byte	Value
0	1
1	1

   In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0101 --type int64
```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

   Which should return 0x0000000000000000 (Status 0).

2. Use OS command 0 to configure Encoder 1 to output raw mode. Check the iC-MU data sheet for more information.

3. Enable Raw mode.

   To change Encoder 1 to Raw mode, the bytes in 0x1023:1 need to be set like this:

Byte	Value
0	1
1	9

   In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0901 --type int64
```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

   Which should return 0x0000000000000000 (Status 0).

Command 2: Open Loop Field Mode (OLFM)

This command is used to be able to configure the settings of the Open Loop Field Mode (OLFM) profiler. The following configuration can be set by using this OS command:

Parameters related to the profiler of the electrical angle:

- Starting electrical angle [mrad]
- Ending electrical angle [mrad]
- Maximum electrical rotational speed [rad / s]
- Electrical rotational acceleration [rad / s²]

Parameters related to the profiler of the magnetic field length:

- Starting length [per thousand rated current]
- Ending length [per thousand rated current]
- Increasing length rate [per thousand rated current / s]

Once the configurations of OLFM profiler are set, the user can put the drive in the corresponding OLFM op_mode (The op_mode for OLFM operation is -3), and by enabling the operation the Drive will generate a magnetic field according to the settings of its profiler. More information about this mode can be found in [Missing link].

Command request

Values in 0x1023:1 for this command:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	2							
Byte 1	Parameter index							
Byte 2	Parameter value [MSB]							
Byte 3	Parameter value [MSB-1]							
Byte 4	Parameter value [MSB-2]							
Byte 5	Parameter value [LSB]							

Explanation of the parameters

- Parameter index: Selects the parameter that shall be configured.

Value	Parameter and units
0	Starting electrical angle [mrad]
1	Ending electrical angle [mrad]
2	Maximum electrical rotational speed [rad / s]
3	Electrical rotational acceleration [rad / s²]
4	Starting length [per thousand rated current]
5	Ending length [per thousand rated current]
6	Increasing length rate [per thousand rated current / s]

- Parameter value: The value to set the selected parameter to. The maximum size for the value is 32 bits and Byte 5 is the least significant byte (LSB).

Command response

The possible responses of this command:

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command	0							

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
completed without errors and without response								
Command completed with error and without response	2							
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 0**: The command was finished successfully.
- **Status 2**: The command couldn't be executed. The received Parameter index value is unknown.

- **Status 3**: This command doesn't have any command-specific OS error code (this status will only happen when a general OS error code occurs).

Example

For setting the "Starting electrical angle [mrad]" to 1000, the parameter value should be converted to hexadecimal so that it is easier to split between multiple bytes. In this case, 1000 is 0x3E8.

Then we should set the bytes in 0x1023:1 like this:

Byte	Value
0	2
1	0
2	0
3	0
4	0x03
5	0xE8

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0xe80300000002 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

Which should return 0x0000000000000000 (Status 0).

Command 3: HRD streaming

This command is used for streaming data into files. The data written to these files is in binary format.

- Multiple files will be created if necessary, up to a maximum of 5 files.
- Each of the files will have a maximum size of 8032 bytes
- The files are named "hr_dataX.bin", where X is a number starting at 0.

Two actions are supported for this command:

- Configure stream: all previous HRD files are deleted from the filesystem, and the user configures the duration of the stream and the variable to stream. Deleting files can take up to 5 seconds in the worst case scenario.
- Start stream: The stream is started with the previously sent configuration.

Command request

The values in 0x1023:1 for this command are as follows:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	3							
1	-							Action
2	Data index							
3	Duration in ms [MSB]							
4	Duration in ms [LSB]							

Explanation of the parameters:

- Action: select the action to perform.

Value	Action
0	Configure stream
1	Start stream

- Data index: select the data to be streamed. This value is only used when the selected Action is Configure stream (0).

Value	Data	Description
0	Encoder raw data	Record encoder raw data. The encoder should be set up first with OS command 1.
1	System iden-	Record system identification data. The identification

Value	Data	Description
	tification data	procedure should be configured with OS command 15 and started before recording of the data.

- Duration in ms: select the duration in milli-seconds of the stream. This value is only used when the Action is Configure stream (0). The maximum value for this parameter is 10000 ms.

Command response

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (with percentage)	100 - 200							
Command in progress (without percentage)	255							
Command completed without errors and	0							

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
without response								
Command completed with error and response	3		OS error code					

- **Status 100-200**: The command to stream the data to a file (Action is 1) is in progress.
- **Status 255**: The command to configure the stream (Action is 0) is in progress.
- **Status 0**: The command has finished successfully.
- **Status 3**: The command has finished unsuccessfully. The error code will be sent, either as a
  - general OS error code or a
  - command-specific OS error code:

OS error code	Error name	Error description
0	Initialization error	The command failed to initialize.
1	Streaming error	Something failed while streaming was in progress.
2	Duration value	The selected Duration value is larger than 10000 ms.
3	Data index value	The selected Data index is unknown.
4	Action value	The selected Action is unknown.

Example

Streaming encoder raw data into a file for 5 s:

1. Use OS command 0 and OS command 1 to configure the iC-MU to output raw data.
2. Configure the streaming:

   To configure the stream first we need to convert the duration to hexadecimal so that it is easier to split between multiple bytes. In this case, 5000 is 0x1388.

   The bytes in 0x1023:1 should be:

Byte	Value
0	3
1	0
2	0
3	0x13
4	0x88

   In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x8813000003 --type int64
```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

   Which should return 0x0000000000000000 (Status 0).

3. Start the streaming.

   The bytes in 0x1023:1 should be:

Byte	Value
0	3
1	1

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0103 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

Which should return 0x0000000000000000 (Status 0).

Command 4: Motor phase order detection

This command is used to be able to detect the order of the motor phases, it is also the first step that is performed during Offset detection. When this command finishes successfully the found value is automatically written to the object 0x2003:5 Motor phases inverted.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically finished. More information about this operation mode can be found in the Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	4							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Com-	1		Motor					

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
mand completed without errors and with response			phase order					
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The detected motor phase order is written in byte 2.

Value	Motor phase order
0	Normal order
1	Inverted order

- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

To measure the motor phase order of our setup, we would need to:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one. For details please refer to Diagnostics opmode.
2. Execute the OS command.

Set Byte 0 of 0x1023:1 to 4.

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x04 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

- If the Motor phase order is normal, the response would be 0x0000000000000001 (Status 0, Normal motor phase order).
- If the Motor phase order is inverted, the response would be 0x0000000000010001 (Status 1, Inverted motor phase order).

Command 5: Commutation offset detection

This command is used to measure the commutation angle offset which is the second step in the Offset Detection procedure. The configuration of the procedure is taken from the already existing and previously used commutation offset objects in the object dictionary. When this command finishes successfully the found value is automatically written to the object 0x2001:0 Commutation angle offset, and the object 0x2009:1 State is set to OFFSET_VALID.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. More information about this operation mode can be found in Diagnostics opmode.

For details about the different methods for Offset detection, refer to Commutation Offset Detection.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	5							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without errors and with response	1		Commutation angle offset [MSB]	Commutation angle offset [LSB]				
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured angle offset is written in bytes 2 and 3.

- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

To measure the commutation angle offset of our setup, we would need to:

1. Configure the Commutation offset detection parameters in the object 0x2009.
2. Switch to diagnostics mode, enter operation enable and disengage the brake if necessary. For details please refer to Diagnostics opmode.
3. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 5. In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x05 --type int64
   ```

   To fetch the response, enter the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

   In case the Commutation angle offset is 2035, this would return 0x0000000007F30001 (Status 1, 2035 is 0x7F3 in hexadecimal).

Command 6: Open phase detection

This command is used for detecting if one of the phases is open.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. For details please refer to Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	6							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without errors and without response	0							
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 0**: The command was finished successfully, no open motor phase was detected.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Open terminal A	Terminal A of the Drive is not connected
1	Open terminal B	Terminal B of the Drive is not connected
2	Open terminal C	Terminal C of the Drive is not connected
3	Open FET A high	Upper FET in leg A of the Drive is not conducting (open circuit fault)
4	Open FET A low	Lower FET in leg A of the Drive is not conducting (open circuit fault)
5	Open FET B high	Upper FET in leg B of the Drive is not conducting (open circuit fault)
6	Open FET B low	Lower FET in leg B of the Drive is not conducting (open circuit fault)
7	Open FET C high	Upper FET in leg C of the Drive is not conducting (open circuit fault)
8	Open FET C low	Lower FET in leg C of the Drive is not conducting (open circuit fault)

Example

To detect if any of the motor phases is open in our setup, we would need to:

1. Switch to diagnostics mode and enter operation enable.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 6. In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x06 --type int64
   ```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

- An OS error code will indicate if there is an issue
- For example if terminal B is open this would return 0x0000000000010003 (Status 3, OS error code 1).
- If no issue was found, this would return 0x0000000000000000 (Status 0).

Command 7: Pole pair detection

This command is used to detect the amount of pole pairs that the connected motor has.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. For details please refer to Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	7							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without	1		Number of pole pairs					

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
errors and with response								
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured number of motor pole pairs is written in byte 2.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Current ampl. error	Drive is not able to increase the amplitude of motor phase currents to the required value. This problem can happen due to limited V_{DC} or high motor phase impedances

Example

For measuring the motor phase order:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 7. In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x07 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

In case the motor has 3 pole pairs, this would return 0x0000000000030001 (Status 1).

Command 8: Phase resistance measurement

This command is used to measure the phase resistance of the motor in mOhm.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. For details please refer to Diagnostics opmode.

Command request

Values in 0x1023:1 for this command:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	8							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without	1		Phase resistance [MSB]	Phase resistance [MSB-	Phase resistance [MSB-	Phase resistance [LSB]		

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
errors and with response				1]	2]			
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured motor phase resistance is written in bytes 2, 3, 4 and 5.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Current ampl. error	Unable to increase the current amplitude.

Example

To measure the motor phase resistance of our setup, we would need to:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 8. In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x08 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

In case the motor phase resistance is 100000, this would return 0x0000000186A00001 (Status 1, 100000 is 0x186A0 in hexadecimal).

Command 9: Phase inductance measurement

This command is used to measure the phase inductance of the motor in uH.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. More information about this operation mode can be found in the Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	9							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Com-	1		Phase	Phase	Phase	Phase		

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
mand completed without errors and with response			inductance [MSB]	inductance [MSB-1]	inductance [MSB-2]	inductance [LSB]		
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured motor phase inductance is written in bytes 2, 3, 4 and 5.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Current ampl. error	Unable to increase the current amplitude.

Example

To measure the motor phase inductance of our setup, we would need to:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 9. In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x09 --type int64
   ```

   To fetch the response, enter the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

   In case the motor phase inductance is 4000, this would return 0x0000000010940001 (Status 1, 4244 is 0x1094 in hexadecimal).

Command 10: Torque constant measurement

This command is used to measure the torque constant of the motor in mNm/A $_{RMS}$.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. More information about this operation mode can be found in Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	0x0A							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without	255							

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
percentage)								
Command completed without errors and with response	1		Torque constant [MSB]	Torque constant [MSB-1]	Torque constant [MSB-2]	Torque constant [LSB]		
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured motor phase inductance is written in bytes 2, 3, 4 and 5.
- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

For measuring the motor torque constant:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one. For details please refer to Diagnostics opmode.

2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 10 (0xA). In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x0A --type int64
   ```

   To fetch the response, enter the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

   In case the motor torque constant is 1203210, this would return 0x000000125C0A0001 (Status 1, 1203210 is 0x125C0A in hexadecimal).

Command 13: Skipped cycles counter

This command is used to check the number of firmware cycles skipped. It provides the value of an internal counter that keeps track of the number of skipped cycles.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	0x0D							
Byte 1	Firmware service							

Explanation of the parameters:

- Firmware service: Select in which service the action should be performed.

Value	Firmware Service
0	Drive control service
1	Motion control service

Command response

The possible responses that this command will use are the following:

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255	-	-	-	-	-	-	-
Command completed without errors and with response	1	-	Number of skipped cycles [MSB]	Number of skipped cycles [MSB-1]	Number of skipped cycles [MSB-2]	Number of skipped cycles [LSB]	-	-
Command completed with error and response	3	-	OS error code	-	-	-	-	-

- Status 255: The command is in progress.
- Status 1: The command finished successfully. The number of skipped cycles is written in bytes 2, 3, 4 and 5.
- Status 3: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

To get the value of the skipped cycles counter of Motion control service:

1. Set Byte 0 of 0x1023:1 to 13 (0xD), and Byte 1 to 1.

   In the terminal we can use the following command:

   ```
   ethercat download 0x1023 1 0x010D --type int64
   ```

   To fetch the response, we should type the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

This will return a value, for example 23. If we run the command again after some time and we get for example 33, this will mean that 10 cycles were skipped in the timespan between both OS commands.

Command 14: Ignore BiSS status bits

This command is used to ignore the BiSS status bits, so that they don't raise a firmware warning or error.

In the case of Circulo, it also stops reading the iC-PVL STATUS register when multiturn is enabled.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	0x0E							
Byte 1						Encoder ordinal		Trigger

Explanation of the parameters:

- Trigger: a value of 1 will ignore the status bits. A value of 0 will stop ignoring them, which is the default behavior of the firmware.
- Encoder ordinal: select the target encoder for the command. A value of 1 will select Encoder 1 (which is the encoder configured in 0x2110: Encoder 1 configuration), a value of 2 will select Encoder 2 (which is the encoder configured in 0x2112: Encoder 2 configuration).

Value	Encoder
1	Encoder 1
2	Encoder 2

Command response

The possible responses that this command will use are the following:

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without errors and without response	0							
Command com-	3		OS error code					

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
pleted with error and response								

- Status 255: The command is in progress.
- Status 0: The command finished successfully.
- Status 3: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

If we want to ignore the BiSS status bits of Encoder 1 we should:

1. Set Byte 0 of 0x1023:1 to 14 (0xE), and Byte 1 to 3 (0b11). In the terminal we can use the following command:

   ```
   ethercat download 0x1023 1 0x030E --type int64
   ```

   To fetch the response, we should type the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

Parameters

Object ID	Name	Description
0x1023	OS command	Selects, starts and reports the status of the command that should be used
0x1024	OS command mode	Orders the next command to be processed or aborts all commands

Quick stop

Overview

When the drive is in motion and the user wishes to stop quickly for any reason, the Quick Stop state can be activated. This function uses a profile to drive the motor from the current velocity down to zero with a specific Quick Stop deceleration. Quick stop can also be performed when an internal fault happens to stop the drive in a controlled way.

> Attention:
> Note that the brake might get damaged when the motor does not ramp down to zero velocity. This can happen due to one of the following reasons:

- when the motor is mechanically unstable
- when the control system is unstably or poorly tuned
- when the quick-stop feature is misconfigured

If no brake exists, this can even lead to a damaged motor!

> Note:
> At the end of the quick stop procedure, the brake gets engaged when the drive is in SWITCH_ON_DISABLED (mode 2). When the drive stays in QUICK_STOP_ACTIVE (mode 6) the brake is not engaged by default.

Usage

Quickstop will be enabled if one of the following conditions occurs:

- Bit 2 (quick stop bit) of the Controlword changed from 1 to 0 (transitions 7, 10, 11 according to the drive state machine diagram.
- A drive fault is triggered
- The drive dropped from EtherCAT Operational state while it was in CiA402 Operational Enabled state

Object 0x605A Quick stop option code has different options that switch the drive into different states:

Option code	Definition
0	Disable drive function (no quick stop deceleration)
2 (default)	Slow down on quick stop ramp and transit into switch on disabled
6	Slow down on quick stop ramp and stay in quick stop active

The execution of the procedure depends on the operational mode that is active:

Position and velocity control modes

- In position control modes (Cyclic Synchronous Position mode, Profile Position mode, Cogging recording, Homing) the quick stop procedure uses the position controller to stop the motor.
- In velocity control modes (Profile velocity mode, Cyclic Synchronous Velocity Mode), the quick stop procedure uses the velocity profiler to stop the motor.

During the procedure, torque offset can be used to compensate for possible torque from the load.

The steps vary depending on the quick stop option code and when quick stop is triggered:

Option code: 0 (Disable the drive)

> Attention:
> In this case, the drive does not perform the quick stop deceleration ramp and is instead immediately disabled.

Option code: 2 (Slow down on quick stop ramp and transit into switch on disabled)

1. The Drive commands a quick stop ramp that ramps down an effective commanded velocity according to the user-defined quick stop deceleration.
2. The end of the quick stop ramp is a moment when the commanded velocity is equal to 0.
3. A timeout period of 100 ms is started in which the drive commands a fixed position or zero velocity.

4. After the end of the quick stop ramp, the drive is disabled, which automatically engages the brake as defined by the brake settings.

Option code: 6 (Slow down on quick stop ramp and stay in quick stop active)

1. The Drive commands a quick stop ramp that ramps down an effective commanded velocity according to the user-defined quick stop deceleration.
2. The end of the quick stop ramp is a moment when the commanded velocity is equal to 0.
3. After the end of the quick stop ramp, the drive continues to stay in Quick Stop Active until a command to disable the drive is received. In position control modes (Cyclic Synchronous Position mode, Profile Position mode, Cogging recording, Homing), Quick Stop Active drive keeps the current position.

   In velocity control modes (Profile velocity mode, Cyclic Synchronous Velocity Mode), the drive controls 0 velocity.

> Attention:
>
> The brake is not engaged by default in this state.

Torque control mode

In torque modes (Cyclic Synchronous Torque mode, Profile torque mode) the quick stop procedure does not use a profile. Instead, the drive commands 0 torque from the start of the quick stop procedure. The steps vary depending on the quick stop option code and when quick stop is triggered:

Option code: 0 (Disable the drive)

> Note:
>
> In this case, the drive does not perform the quick stop deceleration ramp and is instead immediately disabled.

Option code: 2 (Slow down on quick stop ramp and transit into switch on disabled)

1. The Drive commands Target and offset torque
2. At the same moment, the timeout period starts. The timeout period is calculated as actual motor velocity divided by the quick stop deceleration. In this way, a user can configure the maximum duration of the quick stop procedure.
3. During the timeout period, the drive monitors if the drive reaches zero velocity
4. The brake gets engaged and the drive is set to state Switch On Disabled, if one of the conditions are true:
    - The actual velocity of the motor reached 0
    - The timeout period has expired

Option code: 6 (Slow down on quick stop ramp and stay in quick stop active)

1. The Drive commands Target and offset torque. According to the DS 402 standard, the drive remains in the torque control mode it was triggered from.
2. The drive continues to stay in Quick Stop Active until a command to disable the drive is received.

> Note:
> When quick stop is commanded during Offset detection, the procedure is aborted and the drive is disabled.

Parameters

Object ID	Name	Description
0x605A	Quick stop option code	Select the action that is performed when the quick stop function is executed
0x6085	Quick stop deceleration	Defines the quick stop ramp
0x6040	Controlword	Shows the drive's transition into a different state. Use this object to monitor the drive's behavior

Object ID	Name	Description
0x60A9	SI unit velocity	This object defines the velocity value
0x60FC	Position demand internal value	Provides the output of the trajectory generator in profile position mode
0x6064	Position actual value	Provides the actual value of the position encoder
0x606B	Velocity demand value	Output of position controller in position mode/trajectory generator in velocity mode
0x606C	Velocity actual value	Provides the value read at the velocity sensor (or the derivation of the position signal)
0x6074	Torque demand	Provides the output value of the torque trajectory generator
0x6077	Torque actual value	Provides the actual value of the torque

CiA 402 Object Dictionary

All Objects	A list of the entire CiA 402 Object Dictionary
Communication Area	Objects of the Communication Area
Standard RxPDO mapped Objects	The Standard mapping of Receive Process Data Objects
Standard TxPDO mapped Objects	The Standard mapping of Transmit Process Data Objects
Manufacturer Specific Area	A list of all Manufacturer Specific Objects
Profile Specific Area	A list of all Profile Specific Objects

List of all Objects

Table Caption Outside Table: Communication Area

Index		Name
0x1000		Device type
0x1001		Error register
0x1005		COB-ID SYNC
0x1006		Communication cycle period
0x1008		Manufacturer device name
0x100A		Manufacturer software version
0x100C		Guard time
0x100D		Life time factor
0x1010		Store parameters
	0x1010:1	Save all parameters
0x1011		Restore default parameters
	0x1011:1	Restore all default parameters
	0x1011:2	Restore communication default parameters (disabled)
	0x1011:3	Restore application default parameters (disabled)
	0x1011:4	Restore manufacturer default parameters
0x1016		Consumer heartbeat time
	0x1016:1	Consumer heartbeat time
0x1017		Producer heartbeat time
0x1018		Identity object

	0x1018:1	Vendor-ID
	0x1018:2	Product Code
	0x1018:3	Revision number
	0x1018:4	Serial number
0x1019		Synchronous counter overflow value
0x1023		OS command
	0x1023:1	Command
	0x1023:2	Status
	0x1023:3	Response
0x1024		OS command mode
0x1400		Receive PDO1 parameter
	0x1400:1	COB-ID used by PDO
	0x1400:2	Transmission type
	0x1400:3	Inhibit time
0x1401		Receive PDO2 parameter
	0x1401:1	COB-ID used by PDO
	0x1401:2	Transmission type
	0x1401:3	Inhibit time
0x1402		Receive PDO3 parameter
	0x1402:1	COB-ID used by PDO

	0x1402:2	Transmission type
	0x1402:3	Inhibit time
0x1403		Receive PDO4 parameter
	0x1403:1	COB-ID used by PDO
	0x1403:2	Transmission type
	0x1403:3	Inhibit time
0x1800		Transmit PDO1 parameter
	0x1800:1	COB-ID used by PDO
	0x1800:2	Transmission type
	0x1800:3	Inhibit time
	0x1800:4	reserved
	0x1800:5	Event timer
0x1801		Transmit PDO2 parameter
	0x1801:1	COB-ID used by PDO
	0x1801:2	Transmission type
	0x1801:3	Inhibit time
	0x1801:4	reserved
	0x1801:5	Event timer
0x1802		Transmit PDO3 parameter
	0x1802:1	COB-ID used by PDO

	0x1802:2	Transmission type
	0x1802:3	Inhibit time
	0x1802:4	reserved
	0x1802:5	Event timer
0x1803		Transmit PDO4 parameter
	0x1803:1	COB-ID used by PDO
	0x1803:2	Transmission type
	0x1803:3	Inhibit time
	0x1803:4	reserved
	0x1803:5	Event timer
0x1C00		Sync Manager
	0x1C00:1	SyncMan 0
	0x1C00:2	SyncMan 1
	0x1C00:3	SyncMan 2
	0x1C00:4	SyncMan 3
0x1C10		SM 0 Assignment
0x1C11		SM 1 Assignment
0x1C12		SM 2 Assignment
	0x1C12:1	SubIndex 001
	0x1C12:2	SubIndex 002

	0x1C12:3	SubIndex 003
	0x1C12:4	SubIndex 004
0x1C13		SM 3 Assignment
	0x1C13:1	SubIndex 001
	0x1C13:2	SubIndex 002
	0x1C13:3	SubIndex 003
	0x1C13:4	SubIndex 004

Table Caption Outside Table: Standard RxPDO mapped Objects

Index		PDO Index	Name
0x1600			Receive PDO1 mapping
	0x1600:1	0x6040	Controlword
	0x1600:2	0x6060	Modes of operation
	0x1600:3	0x6071	Target Torque
	0x1600:4	0x607A	Target position
	0x1600:5	0x60FF	Target velocity
	0x1600:6	0x60B2	Torque offset
	0x1600:7	0x2701	Tuning command
0x1601			Receive PDO2 mapping
	0x1601:1	0x60FE	Physical outputs
	0x1601:2	0x60FE	Bit mask

0x1602			Receive PDO3 mapping
	0x1602:1	0x2703	User MOSI
	0x1602:2	0x60B1	Velocity offset
0x1603			Receive PDO4 mapping

Table Caption Outside Table: Standard TxPDO mapped Objects

Index		PDO Index	Name
0x1A00			Transmit PDO1 mapping
	0x1A00:1	0x6041	Statusword
	0x1A00:2	0x6061	Modes of operation display
	0x1A00:3	0x6064	Position actual value
	0x1A00:4	0x606C	Velocity actual value
	0x1A00:5	0x6077	Torque actual value
0x1A01			Transmit PDO2 mapping
	0x1A01:1	0x2401	Analog input 1
	0x1A01:2	0x2402	Analog input 2
	0x1A01:3	0x2403	Analog input 3
	0x1A01:4	0x2404	Analog input 4
	0x1A01:5	0x2702	Tuning status
0x1A02			Transmit PDO3 mapping
	0x1A02:1	0x60FD	Digital inputs

0x1A03			Transmit PDO4 mapping
	0x1A03:1	0x2704	User MISO
	0x1A03:2	0x20F0	Timestamp
	0x1A03:3	0x60FC	Position demand internal value
	0x1A03:4	0x606B	Velocity demand value
	0x1A03:5	0x6074	Torque demand

Table Caption Outside Table: Manufacturer Specific Area

Index		Name
0x2000		Command object (disabled)
0x2001		Commutation angle offset
0x2002		Position control strategy
0x2003		Motor specific settings
	0x2003:1	Pole pairs
	0x2003:2	Torque constant
	0x2003:3	Phase resistance
	0x2003:4	Phase inductance
	0x2003:5	Motor phases inverted
0x2004		Brake options
	0x2004:1	Pull voltage

	0x2004:2	Hold voltage
	0x2004:3	Pull time
	0x2004:4	Release strategy
	0x2004:5	Controller disable delay
	0x2004:6	DC bus voltage (deprecated)
	0x2004:7	Brake status
	0x2004:8	Minimum displacement (pin brake)
	0x2004:9	Maximum torque (pin brake)
	0x2004:10	Output voltage
	0x2004:11	Switching frequency
0x2006		Protection
	0x2006:1	Undervoltage setpoint
	0x2006:2	Overvoltage setpoint
	0x2006:3	Overcurrent setpoint
0x2008		Cogging torque compensation
	0x2008:1	State
	0x2008:2	Enabled
0x2009		Commutation offset
	0x2009:1	State
	0x2009:2	Applied percent of rated torque

	0x2009:3	Measurement method
	0x2009:4	Phasing controller Kp
	0x2009:5	Phasing controller Ki
	0x2009:6	Phasing controller Kd
0x200A		I2t
	0x200A:1	Enabled
	0x200A:2	Peak time
	0x200A:3	Percent max
	0x200A:4	Protection active
0x200B		Max power
0x2010		Torque controller
	0x2010:1	Controller Kp (disabled)
	0x2010:2	Controller Ki (disabled)
	0x2010:3	Controller Kd (disabled)
	0x2010:4	Field weakening enabled
	0x2010:5	Field weakening percentage
	0x2010:6	Field weakening starting speed
	0x2010:7	Field weakening ending speed
	0x2010:8	Commutation angle measurement delay
	0x2010:9	Switching frequency

	0x2010:10	Settling time
	0x2010:11	Damping ratio
	0x2010:12	Encoder source
0x2011		Velocity controller
	0x2011:1	Controller Kp
	0x2011:2	Controller Ki
	0x2011:3	Controller Kd
	0x2011:4	Controller integral limit
	0x2011:5	Encoder source
0x2012		Position controller
	0x2012:1	Position loop Kp
	0x2012:2	Position loop Ki
	0x2012:3	Position loop Kd
	0x2012:4	Position loop integral limit
	0x2012:5	Velocity loop Kp
	0x2012:6	Velocity loop Ki
	0x2012:7	Velocity loop Kd
	0x2012:8	Velocity loop integral limit
	0x2012:9	Encoder source
0x2013		Gain scheduling

	0x2013:1	Enabled
	0x2013:2	Threshold velocity 0
	0x2013:3	Position loop Kp 0
	0x2013:4	Position loop Ki 0
	0x2013:5	Position loop Kd 0
	0x2013:6	Position loop integral limit 0
	0x2013:7	Velocity loop Kp 0
	0x2013:8	Velocity loop Ki 0
	0x2013:9	Velocity loop Kd 0
	0x2013:10	Velocity loop integral limit 0
	0x2013:11	Threshold velocity 1
	0x2013:12	Position loop Kp 1
	0x2013:13	Position loop Ki 1
	0x2013:14	Position loop Kd 1
	0x2013:15	Position loop integral limit 1
	0x2013:16	Velocity loop Kp 1
	0x2013:17	Velocity loop Ki 1
	0x2013:18	Velocity loop Kd 1
	0x2013:19	Velocity loop integral limit 1
0x2014		Torque window

	0x2014:1	Value
	0x2014:2	Time
0x2015		Velocity feed forward
	0x2015:1	Gain
	0x2015:2	Cutoff frequency
0x2021		Velocity feedback filter
	0x2021:1	Type
	0x2021:2	Cutoff frequency
0x2022		Position feedback filter
	0x2022:1	Type
	0x2022:2	Cutoff frequency
0x2023		Notch filter
	0x2023:1	Enabled
	0x2023:2	Center frequency
	0x2023:3	Rejection band
0x2027		Control input FIR filter
	0x2027:1	Enabled
	0x2027:2	Order
0x2030		Core temperature
	0x2030:1	Measured temperature

0x2031		Drive temperature
	0x2031:1	Measured temperature
0x2038		External scaled measurement
	0x2038:1	Scaled measurement value
	0x2038:2	Analog input
	0x2038:3	Resistance
	0x2038:4	Constant a0
	0x2038:5	Constant a1
	0x2038:6	Constant a2
	0x2038:7	Constant a3
	0x2038:8	Constant a4
	0x2038:9	Constant a5
	0x2038:10	Cutoff frequency
	0x2038:11	Upper error threshold
	0x2038:12	Lower error threshold
0x203F		Error report
	0x203F:1	Description
0x2040		Input counter
0x20E1		High resolution data
	0x20E1:1	Data 1

	0x20E1:2	Data 2
	0x20E1:3	Data 3
	0x20E1:4	Data 4
0x20F0		Timestamp
0x20F2		Assigned name
0x20F3		DC synchronization
	0x20F3:1	DC pulse time
0x2110		Encoder 1 configuration
	0x2110:1	Sensor port
	0x2110:2	Type
	0x2110:3	Resolution
	0x2110:4	Zero velocity threshold
	0x2110:5	Polarity
	0x2110:6	Singleturn offset
	0x2110:7	Access signal type
	0x2110:8	Clock frequency
	0x2110:9	Frame size
	0x2110:10	Multiturn bits
	0x2110:11	Multiturn first bit position
	0x2110:12	Singleturn bits

	0x2110:13	Singleturn first bit position
	0x2110:14	Timeout
	0x2110:15	CRC polynomial
	0x2110:16	Maximum tbusy
	0x2110:17	Status bits active value
	0x2110:18	Parity type
	0x2110:19	First clock delay
	0x2110:20	Data ordering
	0x2110:21	Endianness
	0x2110:22	Index availability
	0x2110:23	Hall sensor port
	0x2110:24	Sinewave cycles per revolution
	0x2110:25	Sinewave resolution
	0x2110:26	Sinusoidal output voltage
	0x2110:27	Filter
0x2111		Encoder 1 feedback
	0x2111:1	Raw position
	0x2111:2	Adjusted position
	0x2111:3	Velocity
0x2112		Encoder 2 configuration

	0x2112:1	Sensor port
	0x2112:2	Type
	0x2112:3	Resolution
	0x2112:4	Zero velocity threshold
	0x2112:5	Polarity
	0x2112:6	Singleturn offset
	0x2112:7	Access signal type
	0x2112:8	Clock frequency
	0x2112:9	Frame size
	0x2112:10	Multiturn bits
	0x2112:11	Multiturn first bit position
	0x2112:12	Singleturn bits
	0x2112:13	Singleturn first bit position
	0x2112:14	Timeout
	0x2112:15	CRC polynomial
	0x2112:16	Maximum tbusy
	0x2112:17	Status bits active value
	0x2112:18	Parity type
	0x2112:19	First clock delay
	0x2112:20	Data ordering

	0x2112:21	Endianness
	0x2112:22	Index availability
	0x2112:23	Hall sensor port
	0x2112:24	Sinewave cycles per revolution
	0x2112:25	Sinewave resolution
	0x2112:26	Sinusoidal output voltage
	0x2112:27	Filter
0x2113		Encoder 2 feedback
	0x2113:1	Raw position
	0x2113:2	Adjusted position
	0x2113:3	Velocity
0x2210		GPIO pin configuration
	0x2210:1	Pin 1
	0x2210:2	Pin 2
	0x2210:3	Pin 3
	0x2210:4	Pin 4
	0x2210:5	Pin 5
	0x2210:6	Pin 6
	0x2210:7	Pin 7
0x2211		GPIO output events

	0x2211:1	Timestamp setpoint
0x2212		GPIO input actions
	0x2212:1	Timestamp on rising edge
0x2213		GPIO position trigger
	0x2213:1	Position setpoint
	0x2213:2	Target encoder
	0x2213:3	Pulse width
0x2214		GPIO global options
	0x2214:1	Voltage level
0x2401		Analog input 1
0x2402		Analog input 2
0x2403		Analog input 3
0x2404		Analog input 4
0x2611		Safety digital input diagnostics
	0x2611:1	Input 1
	0x2611:2	Input 2
0x2701		Tuning command
0x2702		Tuning status
0x2703		User MOSI
0x2704		User MISO

0x2705	Setup wizard completed

Table Caption Outside Table: Profile Specific Area

Index	Name
0x603F	Error code
0x6040	Controlword
0x6041	Statusword
0x605A	Quick stop option code
0x6060	Modes of operation
0x6061	Modes of operation display
0x6062	Position demand value
0x6064	Position actual value
0x6065	Following error window
0x6066	Following error time out
0x6067	Position window
0x6068	Position window time
0x606B	Velocity demand value
0x606C	Velocity actual value
0x606D	Velocity window
0x606E	Velocity window time
0x606F	Velocity threshold

0x6070		Velocity threshold time
0x6071		Target Torque
0x6072		Max torque
0x6073		Max current
0x6074		Torque demand
0x6075		Motor rated current
0x6076		Motor rated torque
0x6077		Torque actual value
0x6079		DC link circuit voltage
0x607A		Target position
0x607B		Position range limit
	0x607B:1	Min position range limit
	0x607B:2	Max position range limit
0x607C		Home offset
0x607D		Software position limit
	0x607D:1	Min position limit
	0x607D:2	Max position limit
0x607E		Polarity
0x6080		Max motor speed
0x6081		Profile velocity

0x6083		Profile acceleration
0x6084		Profile deceleration
0x6085		Quick stop deceleration
0x6086		Motion profile type
0x6087		Torque slope
0x6088		Torque profile type
0x6091		Gear ratio
	0x6091:1	Motor revolutions
	0x6091:2	Shaft revolutions
0x6098		Homing method
0x6099		Homing speeds
	0x6099:1	Speed during search for switch
	0x6099:2	Speed during search for zero
0x609A		Homing acceleration
0x60A9		SI unit velocity
0x60B1		Velocity offset
0x60B2		Torque offset
0x60F4		Following error actual value
0x60FA		Control effort
0x60FC		Position demand internal value

0x60FD		Digital inputs
0x60FE		Digital outputs
	0x60FE:1	Physical outputs
	0x60FE:2	Bit mask
0x60FF		Target velocity
0x6502		Supported drive modes
0x6621		Safety statusword
	0x6621:1	Byte 1
	0x6621:2	Byte 2

0x1000 Device type

Identifies the type of device.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Device type	0x1000:0	UDINT	32			131-474		read-only (default)	

0x1001 Error register

This object shall provide error information. The CANopen device maps internal errors into this object. It is a part of an emergency object.

Structure of the error register

Bit	M/O	Meaning
0	M	Generic error
1	O	Current
2	O	Voltage
3	O	Temperature
4	O	Communication error (overrun, error state)
5	O	Device profile specific
6	O	reserved (always 0_b)
7	O	manufacturer-specific

If a specific error occurs the corresponding bit shall be set to 1_b. The generic error bit shall be supported. The other bits may be supported. The generic error shall be signaled at any error situation.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Error register	0x1001:0	USINT	8			0		read-only (default)	

0x1005 COB-ID SYNC

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Data			ping
CO-B-ID SY-NC	0x100-5:0	DI-NT	32			128		read-write	

0x1006 Communication cycle period

Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Com-munic-ation cycle period	0x10-06:0	DI-NT	32			100		read-write	

0x1008 Manufacturer device name

This object shall provide the name of the device as given by the manufacturer.

Name	Index-:Sub	Type	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	U-nit	Acc-ess	PDO Map-ping
Man-ufac-	0x10-08:0	STR-ING	40-0			0		read-only	

turer device name		(50)						(default)	

0x100A Manufacturer software version

This object shall provide the manufacturer software version description.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Manufacturer software version	0x100A:0	STRING (50)	40			0		read-only (default)	

0x100C Guard time

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Guard time	0x100C:0	UINT	16			0		read-write	

0x100D Life time factor

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Life time factor	0x100D:0	USINT	8			0		read-write	

0x1010 Store parameters

Provides the ability to save to permanent storage. Write the value `0x65766173` ("evas", or "save" spelled backwards in ASCII) to start the save procedure.

> **save all parameters** - this will save the current configuration to permanent storage.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Save all parameters	0x1010:1	UDINT	32			0		read-write	

0x1011 Restore default parameters

Provides the ability to load from permanent storage. Write the value `0x64616f6c` ("daol", or "load" spelled backwards in ASCII) to start the restore procedure.

> **restore all default parameters** - load the configuration from the config.csv file, stored on permanent storage

The "default" configuration is defined as the contents of the `config.csv` file.

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Restore all default parameters	0x1011:1	UDINT	32			0		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Restore communication default parameters (disabled)	0x1011:2	UDINT	32			0		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Restore application default parameters (disabled)	0x1011:3	UDINT	32			0		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Restore manufacturer default parameters	0x1011:4	UDINT	32			0		read-write	

0x1016 Consumer heartbeat time

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Consumer heartbeat time	0x1016:1	DINT	32			100		read-write	

0x1017 Producer heartbeat time

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Producer heartbeat time	0x1017:0	UINT	16			1000		read-write	

0x1018 Identity object

Provides information about the device.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Vendor-ID	0x1018:1	UDINT	32			8914		read-only (default)	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Product Code	0x1018:2	UDINT	32			513		read-only (default)	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Revision number	0x1018:3	UDINT	32			16777-2162		read-only (default)	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Serial number	0x1018:4	UDINT	32			0		read-only (default)	

0x1019 Synchronous counter overflow value

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Synchronous counter overflow value	0x1019:0	USINT	8			100		read-write	

0x1023 OS command

undefined

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Command	0x1023:1	OCTET_STRING (8)	64			0		read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Status	0x1023:2	USINT	8			0		read-only (default)	

0: last command completed, no errors, no reply.
1: last command completed, no errors, reply there.
2: last command completed, error, no reply.
3: last command completed, error, reply there.
100-200: indicates how much of the command has been executed (in %, 100 = 0%, 200 = 100%).
255: command is executing (if the percentage display is not supported).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Response	0x1023:3	OCTET_	64			0		read-only	

		STRING (8)						(default)	
Byte 0: See SubIndex 2. Byte 1: Unused. 2-n: Command specific response data.									

0x1024 OS command mode

This object shall control the OS command execution in the application specific queue. This object represents the most recent command of an application program specific queue.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
OS command mode	0x1024:0	USINT	8			0		write-only	

0x1400 Receive PDO1 parameter

Indicates the communication parameters for the RPDO 1.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used	0x1400:1	UDINT	32			2147484160		read-write	

by PDO									
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1400:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1400:3	UINT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1401 Receive PDO2 parameter

Indicates the communication parameters for the RPDO 2.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1401:1	UDINT	32			2147484416		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1401:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Inhibit time	0x14-01:3	UI-NT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1402 Receive PDO3 parameter

Indicates the communication parameters for the RPDO 3.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1402:1	UDINT	32			2147484672		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission	0x1402:2	USINT	8			1		read-write	

type									
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1402:3	UINT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1403 Receive PDO4 parameter

Indicates the communication parameters for the RPDO 4.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1403:1	UDINT	32			21474-84928		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1403:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1403:3	UINT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1600 Receive PDO1 mapping

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Con-	0x16-	UD-	32			16148-		read-	

trol-word	00:1	INT				07056		write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Mode-s of oper-ation	0x16-00:2	UD-INT	32			16169-04200		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Target Torqu-e	0x16-00:3	UD-INT	32			16180-18320		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Target pos-ition	0x16-00:4	UD-INT	32			16186-08160		read-write	
Name	Index-	Typ-	Bit	Mi-	M-	Defaul-	U-	Acce-	PDO

	:Sub	e	Si-ze	n D-at-a	ax D-at-a	t Data	nit	ss	Map-ping
Target velo-city	0x16-00:5	UD-INT	32			16273-24448		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Torqu-e off-set	0x16-00:6	UD-INT	32			16222-78160		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Tun-ing com-mand	0x16-00:7	UD-INT	32			65437-6992		read-write	

0x1601 Receive PDO2 mapping

Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-	M-ax D-	Default Data	U-nit	Acce-ss	PDO Map-ping

				at-a	at-a				
Phy-sical out-puts	0x16-01:1	UDI-NT	32			16272-59168		read-write	
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Default Data	U-nit	Acce-ss	PDO Map-ping
Bit mas-k	0x16-01:2	UDI-NT	32			16272-59424		read-write	

0x1602 Receive PDO3 mapping

Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Default Data	U-nit	Acce-ss	PDO Map-ping
Use-r MO-SI	0x16-02:1	UDI-NT	32			65450-8064		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Default Data	U-nit	Acce-ss	PDO Map-ping

Velocity offset	0x1602:2	UDINT	32			16222-12640		read-write	

0x1603 Receive PDO4 mapping

0x1800 Transmit PDO1 parameter

Indicates the communication parameters for the TPDO 1.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1800:1	UDINT	32			21474-84032		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission	0x1800:2	USINT	8			1		read-write	

type									
A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1800:3	UINT	16			1000		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1800:4	USINT	8			0		read-only (default)	
This subindex has no function and exists only for compatibility reasons.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Event timer	0x18-00:5	UI-NT	16			0		read-write	
This time (in ms) can be used to trigger an Event which handles the copying of the data and the sending of the PDO.									

0x1801 Transmit PDO2 parameter

Indicates the communication parameters for the TPDO 2.

Name	Inde-x:Su-b	Typ-e	Bi-t Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PD-O Map-ping
COB-ID used by PDO	0x18-01:1	UD-INT	32			21474-84288		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Inde-x:Su-b	Typ-e	Bi-t Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PD-O Map-ping
Trans-mis-sion type	0x18-01:2	USI-NT	8			1		read-write	

A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1801:3	UINT	16			1000		read-write	

This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1801:4	USINT	8			0		read-only (default)	

This subindex has no function and exists only for compatibility reasons.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Event	0x18-	UI-	16			0		read-	

timer	01:5	NT						write	
This time (in ms) can be used to trigger an Event which handles the copying of the data and the sending of the PDO.									

0x1802 Transmit PDO3 parameter

Indicates the communication parameters for the TPDO 3.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1802:1	UDINT	32			2147484544		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1802:2	USINT	8			1		read-write	

A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1802:3	UINT	16			1000		read-write	

This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1802:4	USINT	8			0		read-write	

This subindex has no function and exists only for compatibility reasons.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Event timer	0x1802:5	UINT	16			0		read-write	

This time (in ms) can be used to trigger an Event which handles the copying of the data and

the sending of the PDO.

0x1803 Transmit PDO4 parameter

Indicates the communication parameters for the TPDO 4.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1803:1	UDINT	32			2147484800		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1803:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1803:3	UINT	16			1000		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1803:4	USINT	8			0		read-write	
This subindex has no function and exists only for compatibility reasons.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Event timer	0x1803:5	UINT	16			0		read-write	
This time (in ms) can be used to trigger an Event which handles the copying of the data and the sending of the PDO.									

0x1A00 Transmit PDO1 mapping

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Statusword	0x1A00:1	UDINT	32			16148-72592		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Modes of operation display	0x1A00:2	UDINT	32			16169-69736		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position actual value	0x1A00:3	UDINT	32			16171-66368		read-write	

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity actual value	0x1A-00:4	UDINT	32			16176-90656		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque actual value	0x1A-00:5	UDINT	32			16184-11536		read-write	

0x1A01 Transmit PDO2 mapping

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 1	0x1A-01:1	UDINT	32			60404-5328		read-write	
Na-	Index-	Typ-	Bit	Mi-	M-	Defaul-	U-	Acce-	PDO

me	:Sub	e	Si-ze	n Da-ta	ax Da-ta	t Data	nit	ss	Map-ping
Ana-log inp-ut 2	0x1A-01:2	UDI-NT	32			60411-0864		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Ana-log inp-ut 3	0x1A-01:3	UDI-NT	32			60417-6400		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Ana-log inp-ut 4	0x1A-01:4	UDI-NT	32			60424-1936		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Tun-	0x1A-	UDI-	32			65444-		read-	

ing stat-us	01:5	NT				2528		write	

0x1A02 Transmit PDO3 mapping

Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Default Data	U-nit	Acce-ss	PDO Map-ping
Dig-ital inp-uts	0x1A-02:1	UDI-NT	32			16271-93376		read-write	

0x1A03 Transmit PDO4 mapping

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
User MISO	0x1A-03:1	UDI-NT	32			65457-3600		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Time-stamp	0x1A-03:2	UDI-NT	32			55259-9584		read-write	

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position demand internal value	0x1A-03:3	UDINT	32			16271-27840		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity demand value	0x1A-03:4	UDINT	32			16176-25120		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque	0x1A-03:5	UDINT	32			16182-14928		read-write	

dema-nd									

0x1C00 Sync Manager

Nam-e	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Syn-cMan 0	0x1C-00:1	USI-NT	8			1		read-only (defa-ult)	
Nam-e	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Syn-cMan 1	0x1C-00:2	USI-NT	8			2		read-only (defa-ult)	
Nam-e	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Syn-cMan 2	0x1C-00:3	USI-NT	8			3		read-only (defa-ult)	

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SyncMan 3	0x1C-00:4	USINT	8			4		read-only (default)	

0x1C10 SM 0 Assignment

0x1C11 SM 1 Assignment

0x1C12 SM 2 Assignment

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 001	0x1C-12:1	UINT	16			563-2		read-write	
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 002	0x1C-12:2	UINT	16			563-3		read-write	
Name	Index:-Sub	Type	Bit Size	Min Da-	Max Da-	Default Dat-	Unit	Access	PDO Mapping

				ta	ta	a			
SubIndex 003	0x1C12:3	UINT	16			5634		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 004	0x1C12:4	UINT	16			5635		read-write	

0x1C13 SM 3 Assignment

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 001	0x1C13:1	UINT	16			6656		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 002	0x1C13:2	UINT	16			6657		read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 003	0x1C13:3	UINT	16			6658		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 004	0x1C13:4	UINT	16			6659		read-write	

0x2000 Command object (disabled)

Command object was used for storing and loading of the configuration from flash memory. This object is substituted with object 0x1010 for storing and object 0x1011 for loading of the configuration. Storing of the configuration is user triggered by entering the value "save" (LittleEndian format) in the object 0x1010, subindex 1. Loading of the configuration loads the default configuration from the flash memory. It is performed during initialization of the slave by entering the value "load" (LittleEndian format) in the object 0x1011, subindex 1.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Command object (dis-	0x2000:0	UINT	16			0		read-write	

abled)									

0x2001 Commutation angle offset

In most of the cases, the position sensor is not physically mounted at the ideal angle on the body of the motor. As a result, an "offset" value should be added to the measured value of position sensor to give the correct (electrical) angle of rotor.

This object represents the required shift for position sensor output. Its range is between [0-4095] and corresponds to [0-360] electrical degrees inside the software.

Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Com-muta-tion angle offset	0x20-01:0	IN-T	16			0		read-write	

0x2002 Position control strategy

Selects the control structure used for position control:

1. Simple PID (A position PID controller directly generating the reference value for the torque controller.)



2. Cascaded PID (A position PID controller that generates the reference value for an underlying velocity controller - while that velocity controller generates the reference value for the torque controller)

image/svg+xml

τ_offset τ_ref τ Cascaded PID

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Position control strategy	0x2002:0	UINT	16	Simple PID	1		read-write	
				Cascaded PID	2			

0x2003 Motor specific settings

Please refer to your motor's data sheet for the specific values.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pole pairs	0x2003:1	USINT	8			1		read-write	
Number of magnetic pole pairs used in the motor. Please refer to your motor datasheet for this value.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque con-	0x2003:2	DINT	32			1	µNm/A	read-write	

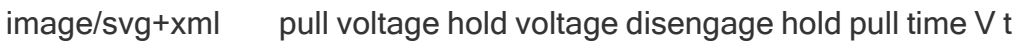
stant									
The motor's torque constant is given by the motor design it defines the torque generated per effective current(rms). The value needs to be configured in µNm/Arms									
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phase resistance	0x2003:3	DINT	32			1	µΩ	read-write	
The resistance between motor terminal and motor neutral point. It is equal to half of "phase to phase resistance" or "line to line resistance".									
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phase inductance	0x2003:4	DINT	32			1	µH	read-write	
The inductance between motor terminal and motor neutral point. It is equal to half of "phase to phase inductance" or "line to line inductance".									
Name	Index-:Sub	Type	Bit Size	Options			Unit	Access	PDO Mapping

Motor phases inverted	0x2003:5	BOOL	1	No 0 Yes 1		read-write	
Use this parameter only when you have a negative velocity or feedback increment with the positive torque increment.							

0x2004 Brake options

Provides options for operating a brake. A brake is spring-activated such that it engages when powered off for safety. It must be disengaged by applying current to a solenoid to be released.

A disengage Pull voltage can be applied to the solenoid to disengage the brake quickly for the duration of the Pull time. After the pull time, a holding voltage is applied to keep the pin retracted.



Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pull voltage	0x2004:1	UDINT	32			0	mV	read-write	
Voltage required to disengage the brake. It is ignored in manual mode.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Hold voltage	0x2004:2	UDINT	32			0	mV	read-write	

Voltage required to hold the brake in a disengaged state. This voltage may be set lower than the "Pull voltage" to reduce losses in the brake. CAUTION! Before setting this parameter, contact your brake manufacturer to ensure reliable operation. It is ignored in manual mode.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pull time	0x2004:3	UINT	16			0	ms	read-write	

Time period in which `0x2004:3 Pull voltage` is applied to release the brake. It is ignored in manual mode.

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Release strategy	0x2004:4	USINT	8	Manual output voltage	0		read-write	
				Clutch-style brake control	1			

				Pin brake control | 2			

Sets the method used to control the brake. The options are:

`Manual output voltage` (0) - Manually control the brake with output voltage.

`Clutch-style brake control` (1) - For brakes that will disengage even if a load is applied to them.

`Pin brake control` (2) - For brakes that require extra motion handling to ensure they release.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller disable delay	0x2004:5	UINT	16			100	ms	read-write	

Time from when the brake is commanded to engage, to when the controller is disabled. It is ignored in manual mode.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
DC bus voltage	0x2004:6	UINT	16			0	V	read-only (default)	

(deprecated)									

Deprecated. Refer to `0x6079 DC link circuit voltage` instead.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Brake status	0x2004:7	USINT	8	Not configured: 0 Engaged: 1 Disengaged: 2		read-write	

Reports the status of the brake. Can also control the brake in automatic mode (Pin brake or Clutch brake)*

The options for controlling or reading this value are:

`Not configured` (0) - No action will be taken for the brake

`Engaged` (1) - The brake will prevent motion (Output voltage disabled in manual mode)

`Disengaged` (2) - The brake will not prevent motion (Output voltage applied to 4th phase in manual mode)

* In Manual Mode, output voltage is controlled by bit 0 of `Digital Output Object (0x60FE)`

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Minimum displacement (pin brake)	0x2004:8	UDINT	32	0	2147483647	0	Inc	read-write	

Minimum angle that the motor needs to turn in before it is certain that the brake pin is pulled out. The units are in commutation sensor increments.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Percentage of Rated Current (pin brake)	0x2004:9	UINT	16	0	200	100		read-write	

Defines the amount of current to be applied during the Pin brake release. Specify this value in percent of rated current.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Output voltage	0x2004:10	UINT	16			0	mV	read-write	Receive PDO (Outputs)
Voltage which is applied on the 4th phase when the Release strategy is set to manual mode (0) and Physical output bit 0 as well as corresponding Bit mask in object `0x60FE` is set to 1.									
Name	Index:Sub	Type	Bit Size	Options			Unit	Access	PDO Mapping
Switching frequency	0x2004:11	USINT	8	16 kHz	0			read-write	
				32 kHz	1				
				64 kHz	2				
Select the switching frequency for the 4th phase (D).									

0x2006 Protection

Provides user-defined values for setting protection thresholds. If the measured value of one of these parameters exceeds the threshold, the drive will issue an error and halt the drive.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Under-	0x20-	DI-	32			0	V	read-	

voltage set-point	06:1	NT						write	
Custom value for undervoltage protection.									
Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Over-voltage set-point	0x20-06:2	DI-NT	32			100-0	V	read-write	
Custom value for overvoltage protection.									
Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Over-current set-point	0x20-06:3	DI-NT	32			500-000	m-A	read-write	
Custom value for overcurrent protection.									

0x2008 Cogging torque compensation

Cogging compensation is a software feature that suppresses torque ripple of electric motors to overcome vibration problems in precision applications. When cogging compensation is activated, an additional torque component is generated actively by the motor in order to compensate the cogging torque. To find the amount of compensation torque needed at each motor position, a recording procedure has to be performed before the compensation itself can be used. This object

is used for reporting the state of the motor cogging data and enabling or disabling the cogging compensation feature.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
State	0x2008:1	INT	16	NO DATA 0 IN PROGRESS 1 SAVING 2 DATA EXISTS 3		read-only (default)	

State of the cogging compensation procedure and data.

Possible values:

`NO DATA` (0) - No cogging data has been stored. Use the Cogging Recording opmode to record new data.
`IN PROGRESS` (1) - Recording of the cogging data is in progress.
`SAVING` (2) - Recording is done and the data is being saved to memory.
`DATA EXISTS` (3) - Cogging data has been loaded from memory. Use the Enabled subfield to enable the data for cogging compensation.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2008:2	INT	16	Disabled 0 Enabled 1		read-write	

Enables or disables the cogging compensation feature.

0x2009 Commutation offset

Used for reporting the state of the commutation offset.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
State	0x2009:1	INT	16	OFFSET INVALID: 0 IN PROGRESS: 1 OFFSET VALID: 2		read-write	

This subitem shows the state of the commutation offset. A new offset can be found by entering the Commutation Offset Detection opmode.

`OFFSET INVALID` (0) - Offset detection needs to be executed before using the torque controller.
`IN PROGRESS` (1) - Commutation Sensor Offset Detection is currently in progress.
`OFFSET VALID` (2) - The offset has been found and is valid. The torque controller can now be used.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Applie-	0x20-	IN-	16	0	100	100		read-	

d percent of rated torque	09:2	T						write	

Defines the amount of torque to be applied during the offset detection process. Specify this value in percent of rated torque.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Measurement method	0x2009:3	SINT	8	0	2	0		read-write	

Used method for measurment of commutation angle offset
Default value is 0. This method (method-0) does not need any tuning, it might rotate the motor up to 100% of a pole-pare range during phasing procedure.
If 1 is selected, method-1 will be executed to find the proper value of commutation angle offset. Before using method-1 the following steps should be completed:

Proper value of "phase inverted" parameter should be updated in the software.
PID parameters of phasing controller should be tuned. A well tuned phasing controller will result in no visible movement during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

If 2 is selected, method-2 will be executed to find the proper value of commutation angle offset. Regarding method 2 the following points should be considered:

Method 2 does not need any movement on motor shaft
The reliability of method 2 is a function of motor design structure. As a result, for high

reliabilty applications it is always recommended to use method 0 or method 1.
If Method 2 is selected, and commutation sensor has an index, then at startup the Drive uses method 2 to find a temporary offset. Once the index is found, this temporary offset will be replaced by the offset which is saved in object dictionary.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phasing controller Kp	0x2009:4	REAL	32	0		0		read-write	

Proportional constant Kp in PID phasing controller. This controller is used to hold the position during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phasing controller Ki	0x2009:5	REAL	32	0		0		read-write	

Integral constant Ki in PID phasing controller. This controller is used to hold the position during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phasing controller Kd	0x2009:6	REAL	32	0		0		read-write	

Derivative constant Kd in PID phasing controller. This controller is used to hold the position during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

0x200A I2t

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Enabled	0x200A:1	USINT	8	Disabled	0		read-write	

				Enabled (current limit at 100%) 1 Enabled (error at 100%) 2					
Enables or disables the I2t protection feature.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Peak time	0x20-0A:2	UDINT	32		360-0000	300-0	ms	read-write	
This field determines how long the motor can operate while peak rms current is flowing in its windings. This time should be determined by motor manufacturer.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Percent max	0x20-0A:3	USINT	8			0		read-only (default)	
This is an index of your motor "over-heat" status. It changes between 0 and 250. At 0%, the motor is not over heated. When this value reaches 100% the motor is at its maximum allowed over heated condition, and if I2t is enabled, the maximum rms current will be limited.									

This limitation is going to be removed when the percentage goes down to 50% or less.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Protection active	0x200A:4	BOOL	1			0		read-only (default)	
As long as it is 0, i2t protection is not putting any limit on maximum rms current of your motor. Once it is 1 it means that i2t protection is limiting the "maximum rms currents" of your motor to its rated value.									

0x200B Max power

This object is a limit on maximum mechanical power on motor shaft. Wrong settings of "motor torque constant", "position sensor configurations" or "number of motor pole-pairs" can lead to misbehavior of power limitation feature. By default the power of each Drive is internally limited to 110% of its Maximum Peak Output Power value. As a result, this object will only be effective if its value is less than 110% of the drive's Maximum Peak Output Power.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max power	0x200B:0	UDINT	32	1	2147483647	2147483647	W	read-write	

0x2010 Torque controller

Range of all torque controller gains is 0 to 100,000,000

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Kp (disabled)	0x2010:1	REAL	32	0		1159479296		read-write	
In the case of using predictive controllers for current loop the unit is undefined, and the default value is 2500. Moreover, Increasing this parameter directly amplifies the amplitude of applied voltages, and as a result has an effect similar to Kp in the case of using a PID controller. Range of this torque controller gain is 0 to 100,000,000.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Ki (disabled)	0x2010:2	REAL	32	0		1193033728		read-write	
The unit of this gain is 0.16 [uV/(Second * Amp_rms)]. The default value is 40000. Range of this torque controller gain is 0 to 100,000,000.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

			ze		a				ping
Con-troller Kd (dis-abled)	0x20-10:3	RE-AL	3-2	0		0		read-write	

This parameter is undefined for predictive current controller.
Range of this torque controller gain is 0 to 100,000,000.

Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Options	U-ni-t	Acc-ess	PD-O Map-ping
Field weak-ening enable-d	0x20-10:4	UD-INT	3-2	Disabled 0 Enabled 1		read-write	

In many applications, it is required to extend the velocity range of an electric motor.

This parameter enables or disables “Field Weakening”, a feature that increases the speed range of a PM motor by weakening the magnetic field of the rotor linearly over the speed range.

The user should set 3 parameters:

1. Field Weakening Starting Speed (Subitem 6)

2. Field Weakening Ending Speed (Subitem7)

3. Percentage of Field Weakening (f) (Subitem 5)

Note:

While the field weakening feature is enabled, it is assumed that the Drive is in its normal conditions with no fault.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Field weakening percentage	0x2010:5	UDINT	32	0	100	0		read-write	
Percentage of Field weakening. This field can have an integer value in the range of [0-100], and determines the "RMS" value of phase currents which are (only) generated to weaken the field. For example, if set to 10, then 10% of rated current will be injected in motor phases to weaken the rotor field. WARNING: Based on motor type, values more than 25% can lead to instability of torque control service. While tuning, proceed with "small steps", "limited power supply voltage" and "limited power supply current".									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Field weakening starting speed	0x2010:6	UDINT	32	0	500000	0	rpm	read-write	

Field weakening is increased linearly from 0% (at starting speed point) to its final value (at ending speed point). The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Field weakening ending speed	0x2010:7	UDINT	32	0	5000000	0	rpm	read-write	

Field weakening is increased linearly from 0% (at starting speed point) to its final value (at ending speed point). The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Commutation angle measurement delay	0x2010:8	UDINT	32			0	µs	read-write	

Time taken by the sensor in reading the position value. An offset angle corresponding to this sensor measurement delay is calculated by the firmware as shown by the equation below.

$$\text{Sensor Delay Angle Offset} = \text{Electrical Rotational Speed} \times \text{Sensor Measurement Delay}$$

This angle offset value is then added into the sampled electrical angle.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Switching frequency	0x2010:9	USINT	8	16 kHz: 0 32 kHz: 1 64 kHz: 2		read-write	

Select the switching frequency for the 3 motor phases (A, B, C).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Settling time	0x2010:10	UINT	16	100	10000	2000	µs	read-write	

Settling time of current loop controllers. This value should be entered in [micro seconds]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Damping ratio	0x2010:11	UINT	16	100	10000	2000		read-write	

Damping ratio of current loop controllers. This value should be entered in [per mill]. As an

<table>
<tr><td colspan="8">example, if a damping ratio of 1.5 is required, then this value should be set to 1500.</td></tr>
<tr><td>Name</td><td>Index:Sub</td><td>Type</td><td>Bit Size</td><td>Options</td><td>Unit</td><td>Access</td><td>PDO Mapping</td></tr>
<tr><td>Encoder source</td><td>0x2010:1-2</td><td>USINT</td><td>8</td><td><table><tr><td>No encoder</td><td>0</td></tr><tr><td>Encoder 1</td><td>1</td></tr><tr><td>Encoder 2</td><td>2</td></tr></table></td><td></td><td>read-write</td><td></td></tr>
<tr><td colspan="8">Select which encoder should be used for torque control. Must be mounted to the motor shaft. Used for commutation.</td></tr>
</table>

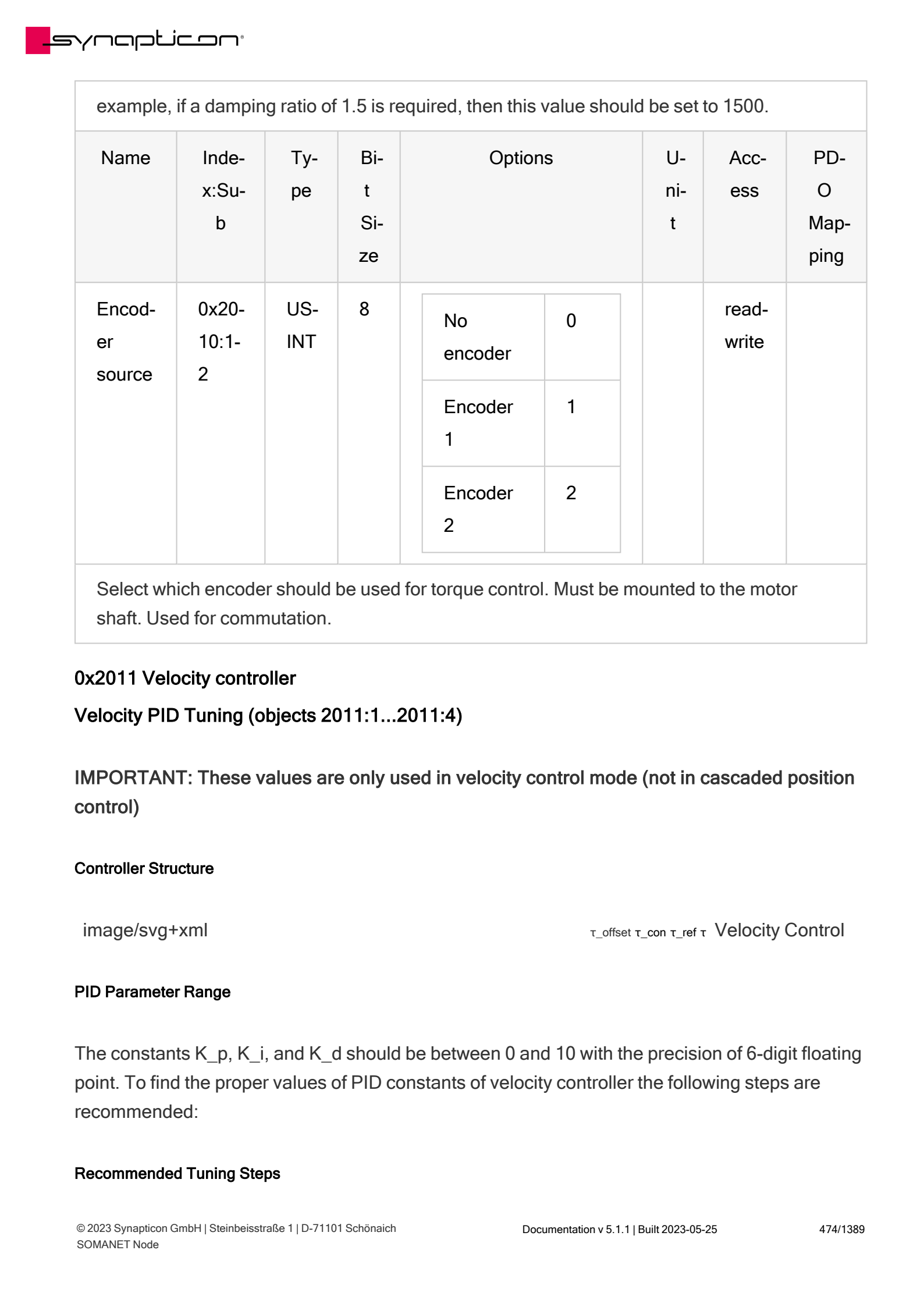
0x2011 Velocity controller

Velocity PID Tuning (objects 2011:1...2011:4)

IMPORTANT: These values are only used in velocity control mode (not in cascaded position control)

Controller Structure

image/svg+xml



PID Parameter Range

The constants K_p, K_i, and K_d should be between 0 and 10 with the precision of 6-digit floating point. To find the proper values of PID constants of velocity controller the following steps are recommended:

Recommended Tuning Steps

- Step 1. Check if all constants K_p, K_i, and K_d are set to 0. After that activate the velocity controller. By default, the integral limit of velocity controller should be set to the maximum torque of selected motor in [mNm].
- Step 2. Set the reference velocity to 50% of rated motor velocity [rpm].
- Step 3. Increase K_p until the measured speed is around 70% of the reference speed. As a suggestion start with K_p equal to 0.1, and in each step increase K_p with a factor of 2.
- Step 4. From this step forth, step response of the system should be checked. Increase K_i step by step to increase the speed of velocity controller (as a suggestion start with K_i equal to 0.01 and in each step increase it by a factor of 2). At this point, the parameters K_p and K_i of the velocity controller are tuned. The parameter K_d of the velocity controller can be kept at its default value (0).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Kp	0x2011:1	REAL	32	0	65535	0		read-write	
The P-gain of the PID Velocity Control Loop [mNm/rpm].									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Ki	0x2011:2	REAL	32	0	65535	0		read-write	
The I-gain of the PID Velocity Control Loop [mNm/(rpm*s)]									
Name	Index:Sub	Type	Bit Si-	Min Dat-	Max	Default	Unit	Access	PDO Map-

			ze	a	Dat-a	Dat-a			ping
Con-troller Kd	0x20-11:3	RE-AL	32	0	655-35	0		read-write	
The D-gain of the PID Velocity Control Loop [rpm*s/inc]									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Min Dat-a	Ma-x Dat-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Con-troller integ-ral limit	0x20-11:4	UDI-NT	32	0		0	m-N-m	read-write	
Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually.									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Options			Un-it	Acce-ss	PDO Map-ping
Enco-der sour-ce	0x20-11:5	USI-NT	8	No encoder	0			read-write	
				Encoder 1	1				

				Encoder 2 | 2			
Select which encoder should be used for velocity control.							

0x2012 Position controller

Parameter Tuning Guide for Position Controller with Cascaded Structure

Introduction

The cascaded control structure is the standard position control structure for SOMANET Drives. Its controller gains can be automatically set by the auto-tuning process, the results will be stored in the objects 2012:1 ... 2012:8. If auto-tuning is not applicable or manual modifications are necessary, the gains can be set manually. In the following, a procedure to do so is described. If a simple PID position controller is selected instead of the cascaded PID position controller, its gains are defined by the objects 2012:1 ... 2012:4.

Control Basics

image/svg+xml

τ_{offset} τ_{ref} τ Cascaded PID

Control Basics

Cascaded controller is usually used in PI-P form, i.e. inner loop (velocity control loop) is only using the proportional part of its PID controller and the outer loop (position controlling loop) is only using its proportional and integral part. The other form is P-PI form, i.e. inner loop is using proportional and integral part and outer loop uses only proportional part. The inner velocity control loop is responsible for calculating the torque reference. As a result the integral limit of the velocity controller should be set to the maximum torque of your torque actuato. Moreover, the outer position control loop is responsible for controlling the reference velocity, and consequently its integral limit should be set to the maximum velocity of your system.

Tuning Concept in Brief

For tuning the cascaded structure, we should first focus on the inner (velocity) loop. As it is the inner controlling loop, it should be faster than the outer position loop. However, if all parameters of the outer position controller are set to 0, the inner loop will be disconnected from the user position commands. As a result, we can increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity` = 10 * K_p_position`). Once the real position started to follow the reference position, we can stop increasing the K_p_position, and only focus on increasing K_p_velocity. At this stage, we can increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is caused by controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise. Once the velocity controller (the inner controlling loop) is tuned, it is time to tune the position controller (the outer controlling loop). To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of too sharpened position control. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. So far, the proportional parts of both inner velocity loop and outer position loop are tuned, and we can focus on integrator part of the outer position or inner velocity loop (depends on chosen structure of controller). Increasing the integrator constant will remove the steady state error, but at the same time, it adds some overshoot at step responses. As a rule of thumb, you can increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

In the following section, the explained tuning concept is divided in separate systematic steps.

Tuning Steps

- Step 1. Set the PID constants of both controllers equal to 0. By default, the integral limit of the velocity controller should be set to motor maximum torque and the integral limit of position controller should be set to motor maximum velocity. **From this step forth, the step response of position controller should always be evaluated.**
- Step 2. In this step, the K_p constant of the velocity controller should be tuned. Increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity = 10 * K_p_position). Once the real position starts to follow the reference position, stop increasing the K_p_

position and only focus on increasing the K_p_velocity. Increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is because of controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise.

- Step 3. Now that the velocity controller is tuned, it is time to tune the parameters of the PID position controller. Start with K_p_position, and increase it until you again get close to the instability margin, and then reduce K_p_position` to its 90%. So far, the proportional parts of both inner velocity loop and outer position loop are tuned.
- Step 4. In this step, increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Parameter Tuning Guide for Position Controller with simple PID Structure (objects 2012:1...2012:4 only)

The Position Controller with Cascaded Structure is generally preferable over the simple PID controller. Apart from academic purposes, there are few reasons to use the simple PID controller.

Control Structure

image/svg+xml

τ_offset τ_ref τ Simple PID

Tuning Concept for Simple PID Controller

To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of a position control that has been sharpened too much. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. Now increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kp	0x2012:1	REAL	32	0	65535	0		read-write	
The P-gain of the PID Position Control Loop. The value is given in [rpm/inc] for the cascaded position controller ([mNm/inc] for the simple PID position controller)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Ki	0x2012:2	REAL	32	0	65535	0		read-write	
The I-gain of the PID Position Control Loop. The value is given in [rpm/(inc*s)] for the cascaded position controller ([mNm/(inc*s)] for the simple PID position controller)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pos-	0x20-	RE-	32	0	655-	0		read-	

ition loop Kd	12:3	AL			35			write	

The D-gain of the Position Control Loop [rpm*s/(inc)] for the cascaded position controller ([mNm*s/(inc)] for the simple PID position controller)

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop integral limit	0x2012:4	UDINT	32	0		0	rpm	read-write	

Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kp	0x2012:5	REAL	32	0	65535	0		read-write	

The P-gain of the PID Velocity Loop Controller [mNm/rpm]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Ki	0x2012:6	REAL	32	0	65535	0		read-write	
The I-gain of the PID Velocity Controller [mNm/(rpm*s)]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kd	0x2012:7	REAL	32	0	65535	0		read-write	
The D-gain of the PID Velocity Loop Controller [mNm*s/rpm]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop integral	0x2012:8	UDINT	32	0		0	mNm	read-write	

limit									
Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually.									
Name	Index:Sub	Type	Bit Size	Options			Unit	Access	PDO Mapping
Encoder source	0x2012:9	USINT	8	No encoder	0			read-write	
				Encoder 1	1				
				Encoder 2	2				
Select which encoder should be used for position control.									

0x2012 Position controller

Parameter Tuning Guide for Position Controller with Cascaded Structure

Introduction

The cascaded control structure is the standard position control structure for SOMANET Drives. Its controller gains can be automatically set by the auto-tuning process, the results will be stored in the objects 2012:1 ... 2012:8. If auto-tuning is not applicable or manual modifications are necessary, the gains can be set manually. In the following, a procedure to do so is described. If a simple PID position controller is selected instead of the cascaded PID position controller, its gains are defined by the objects 2012:1 ... 2012:4.

Control Basics

image/svg+xml τ_offset τ_ref τ Cascaded PID

Control Basics

Cascaded controller is usually used in PI-P form, i.e. inner loop (velocity control loop) is only using the proportional part of its PID controller and the outer loop (position controlling loop) is only using its proportional and integral part. The other form is P-PI form, i.e. inner loop is using proportional and integral part and outer loop uses only proportional part. The inner velocity control loop is responsible for calculating the torque reference. As a result the integral limit of the velocity controller should be set to the maximum torque of your torque actuato. Moreover, the outer position control loop is responsible for controlling the reference velocity, and consequently its integral limit should be set to the maximum velocity of your system.

Tuning Concept in Brief

For tuning the cascaded structure, we should first focus on the inner (velocity) loop. As it is the inner controlling loop, it should be faster than the outer position loop. However, if all parameters of the outer position controller are set to 0, the inner loop will be disconnected from the user position commands. As a result, we can increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity` = 10 * K_p_position`). Once the real position started to follow the reference position, we can stop increasing the K_p_position, and only focus on increasing K_p_velocity. At this stage, we can increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is caused by controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise. Once the velocity controller (the inner controlling loop) is tuned, it is time to tune the position controller (the outer controlling loop). To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of too sharpened position control. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. So far, the proportional parts of both inner velocity loop and outer position loop are tuned, and we can focus on integrator part of the outer position or inner velocity loop (depends on chosen structure of controller). Increasing the integrator constant will remove the steady state error, but at the same time, it adds some overshoot at step responses. As a rule of thumb, you can increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

In the following section, the explained tuning concept is divided in separate systematic steps.

Tuning Steps

- Step 1. Set the PID constants of both controllers equal to 0. By default, the integral limit of the velocity controller should be set to motor maximum torque and the integral limit of position controller should be set to motor maximum velocity. **From this step forth, the step response of position controller should always be evaluated.**
- Step 2. In this step, the K_p constant of the velocity controller should be tuned. Increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity = 10 * K_p_position). Once the real position starts to follow the reference position, stop increasing the K_p_position and only focus on increasing the K_p_velocity. Increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is because of controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise.
- Step 3. Now that the velocity controller is tuned, it is time to tune the parameters of the PID position controller. Start with K_p_position, and increase it until you again get close to the instability margin, and then reduce K_p_position` to its 90%. So far, the proportional parts of both inner velocity loop and outer position loop are tuned.
- Step 4. In this step, increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Parameter Tuning Guide for Position Controller with simple PID Structure (objects 2012:1...2012:4 only)

The Position Controller with Cascaded Structure is generally preferable over the simple PID controller. Apart from academic purposes, there are few reasons to use the simple PID controller.

Control Structure

image/svg+xml τ_offset τ_ref τ Simple PID

Tuning Concept for Simple PID Controller

To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of a position control that has been sharpened too much. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. Now increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kp	0x2012:1	REAL	32	0	65535	0		read-write	
The P-gain of the PID Position Control Loop. The value is given in [rpm/inc] for the cascaded position controller ([mNm/inc] for the simple PID position controller)									

Name	Index:Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Dat-a	Dat-a			ping
Pos-ition loop Ki	0x20-12:2	RE-AL	32	0	655-35	0		read-write	
The I-gain of the PID Position Control Loop. The value is given in [rpm/(inc*s)] for the cas-caded position controller ([mNm/(inc*s)] for the simple PID position controller)									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Dat-a	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Pos-ition loop Kd	0x20-12:3	RE-AL	32	0	655-35	0		read-write	
The D-gain of the Position Control Loop [rpm*s/(inc)] for the cascaded position controller ([mNm*s/(inc)] for the simple PID position controller)									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Dat-a	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Pos-ition loop integ-ral limit	0x20-12:4	UDI-NT	32	0		0	rp-m	read-write	

Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

0x2013 Gain scheduling

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2013:1	BOOL	1	Disabled: 0 Enabled: 1		read-write	
Enables or disables the gain scheduling feature.							

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Threshold velocity 0	0x2013:2	UDINT	32	0		0	rpm	read-write	
Velocity that determines the lower threshold. Underneath this value, controller gains set 0 is used. Beyond it, linear interpolation begins. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).									

Name	Index:Sub	Type	Bit Si-	Min Dat-	Max	Default	Unit	Access	PDO Map-

			ze	a	Dat-a	Dat-a			ping
Pos-ition loop Kp 0	0x20-13:3	RE-AL	32	0	655-35	0		read-write	
The P-gain of the Position Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm/inc]									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Min Dat-a	Ma-x Dat-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Pos-ition loop Ki 0	0x20-13:4	RE-AL	32	0	655-35	0		read-write	
The I-gain of the Position Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm/(inc*s)]									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Min Dat-a	Ma-x Dat-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Pos-ition loop Kd 0	0x20-13:5	RE-AL	32	0	655-35	0		read-write	
The D-gain of the Position Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm*s/inc]									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop integral limit 0	0x2013:6	UDINT	32	0		0	rpm	read-write	
The integrator limit of the Position Control Loop of area 0 (low velocity, below threshold velocity 0). Technique used for preventing the windup is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kp 0	0x2013:7	REAL	32	0	65535	0		read-write	
The P-gain of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0) [mNm/rpm]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

					a	a			
Velocity loop Ki 0	0x2013:8	REAL	32	0	65535	0		read-write	
The I-gain of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0) [mNm/(rpm*s)]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kd 0	0x2013:9	REAL	32	0	65535	0		read-write	
The D-gain of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm*s/inc]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop integral limit 0	0x2013:10	UDINT	32	0		0	mNm	read-write	

The integrator limit of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0). Technique used for preventing the windup is clamping and it prevents the integral term from accumulating above or below pre-determined bounds.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Threshold velocity 1	0x2013:11	UDINT	32	0		0	rpm	read-write	

Velocity that determines the upper threshold. Above this value, controller gains set 1 is used. Up to this velocity linear interpolation is performed. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kp 1	0x2013:12	REAL	32	0	65535	0		read-write	

The P-gain of the Position Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm/inc]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Position loop Ki 1	0x20-13:13	REAL	32	0	655-35	0		read-write	

The I-gain of the Position Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm/(inc*s)]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kd 1	0x20-13:14	REAL	32	0	655-35	0		read-write	

The D-gain of the Position Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm*s/inc]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop integral limit 1	0x20-13:15	UDINT	32	0		0	rpm	read-write	

The integrator limit of the Position Control Loop of area 1 (high velocity, above threshold velocity 1). Technique used for preventing the windup is clamping and it prevents the integ-

ral term from accumulating above or below pre-determined bounds. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kp 1	0x2013:16	REAL	32	0	65535	0		read-write	

The P-gain of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1) [mNm/rpm]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Ki 1	0x2013:17	REAL	32	0	65535	0		read-write	

The I-gain of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1) [mNm/(rpm*s)]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Velocity loop Kd 1	0x20-13:18	REAL	32	0	655-35	0		read-write	
The D-gain of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm*s/inc]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop integral limit 1	0x20-13:19	UDINT	32	0		0	mNm	read-write	
The integrator limit of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1). Technique used for preventing the windup is clamping and it prevents the integral term from accumulating above or below pre-determined bounds.									

0x2014 Torque window

undefined

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Value	0x2014:1	UINT	16			65535		read-write	
This object shall indicate the configured symmetrical range of accepted torque values rel-									

ative to the target torque. If the actual torque is within the torque window, target torque shall be regarded as having been reached. The value shall be given per thousand of rated torque. If the value of the torque window is 0xFFFF, torque window monitoring will be switched off.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Time	0x2014:2	UINT	16			0	ms	read-write	
This object shall indicate the configured time, during which the actual torque within the torque window is measured. The value shall be given in ms.									

0x2015 Velocity feed forward

Velocity feed-forward takes the derivative of the target position and filters the result by using the 1st order low pass filter. The output of the filter is then multiplied by the configured gain.

The end result is added as an additional input to the velocity controller in cascaded position control mode.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Gain	0x2015:1	INT	16	0	5000	0		read-write	
Setting the gain to 0 disables the feedforward feature. Any other value configures the end gain and enables the feature. The value is given in per-mil percent of estimated velocity.									

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Cutoff frequency	0x2015:2	UINT	16	5	2000	20	Hz	read-write	
Cutoff frequency of the 1st order low pass filter that filters the calculated velocity.									

0x2021 Velocity feedback filter

Measured velocity signal from the sensor can be filtered. There is only one type of filtering that the user can choose:
1st order low pass filter with formula G(s)=1/(1+sT), where T=1/(2*pi*f),
The filter is implemented in discrete domain using Euler's forward difference discretization method, i.e. s=1/T*(1-z^-1).

The filter is enabled by setting the filter type parameter to 1, and disabled by setting it to 0. Parameter cutoff frequency is set in [Hz].

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Type	0x2021:1	USINT	8	Disabled	0		read-write	
				1st order low pass	1			

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Cutoff frequency	0x2021:2	UDINT	32			0	Hz	read-write	

0x2022 Position feedback filter

Measured position signal from the sensor can be filtered. There is only one type of filtering that the user can choose:
1st order low pass filter with formula G(s)=1/(1+sT), where T=1/(2*pi*f),
The filter is implemented in discrete domain using Euler's forward difference discretization method, i.e. s=1/T*(1-z^-1).

The filter is enabled by setting the filter type parameter to 1, and disabled by setting it to 0. Parameter cutoff frequency is set in [Hz].

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Type	0x2022:1	USINT	8	Disabled: 0 1st order low pass: 1		read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Cutoff frequency	0x2022:2	UDINT	32			0	Hz	read-write	

0x2023 Notch filter

Implemented as IIR 2nd order filter configured by the center frequency and the bandwidth around it. Filter is placed on the output of the velocity controller so it filters the torque demand value. Frequency prewarping is applied. Filter formula in discrete domain

$y_k = a_0 x_k + a_1 x_{k-1} + a_2 x_{k-2} + b_1 y_{k-1} + b_2 y_{k-2}$

where $a_0 = K$, $a_1 = -2K\cos(2\pi f)$, $a2 = K$, $b1 = 2R\cos(2\pi f)$, $b2 = -R^2$

and K = (1 - 2Rcos(2pi*f) + R^2) / (2-2cos(2pi*f))

R = 1 - 3 * BW

f - center frequency and BW - bandwidth expressed relative to the sampling frequency, where f,BW = [0, 0.5]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2023:1	BOOL	1	Disabled 0 Enabled 1		read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Center frequency	0x2023:2	UINT	16	10	2000	100	Hz	read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Rejection band	0x2023:3	UINT	16	1	4000	1	Hz	read-write	

0x2027 Control input FIR filter

FIR filter (boxcar)

Depending on the mode of operation, this filter acts on the position or velocity demand values sent from the master. The sampling frequency is the same as the cycle time of the position controller (4 kHz). It can be used as an input profiler (or "shaper") to realize linear interpolation between successive commands (see documentation), or to limit the jerk of the profilers (see documentation). The default value is suited for masters operating at 1 kHz.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2027:1	BOOL	1	Disabled 0 Enabled 1		read-write	
Set to 1 to enable the control input FIR filter.							

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Order	0x2027:2	USINT	8	0	31	3		read-write	
The order or length of the FIR filter. Order of the filter means the maximum number of delay elements used in the filter circuit.									

0x2030 Core temperature

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Measured temper-	0x2030:1	DINT	32			0	m°C	read-only (default)	Transmit PDO (Inpu-

ature									ts)
Describes the temperature measured on the Core board.									

0x2031 Drive temperature

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Measured temperature	0x2031:1	DINT	32			0	m°C	read-only (default)	Transmit PDO (Inputs)
Describes the temperature measured near the inverter.									

0x2038 External scaled measurement

Feature that takes an analog input and transforms it to an output as defined by a 5th order polynomial.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Scaled measurement value	0x2038:1	REAL	32			0	1	read-only (default)	Transmit PDO (Inputs)

Scaled value of the selected analog input after applying the selected 5th order polynomial.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Analog input	0x2038:2	USINT	8	Internal: 0 Analog input 1: 1 Analog input 2: 2 Analog input 3: 3 Analog input 4: 4		read-write	

Select the scaled measurement source.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Resistance	0x2038:3	UDINT	32			0	Ω	read-write	

Select the pull-up resistor connected to the selected analog input.
In case there is no pull-up resistor "0" should be selected. In that case, the polynomial would be applied to the ADC count directly (range: 0 - 4095).
Otherwise, it will be assume the user connected a pull-down resistor to create a voltage divider. The pull-down resistor value will be calculated and the polynomial will be applied to

that value.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Constant a0	0x2038:4	REAL	32			0		read-write	
Constant a0 of the polynomial (Offset)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Constant a1	0x2038:5	REAL	32			0		read-write	
Constant a1 of the polynomial (1st order coefficient)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Constant a2	0x2038:6	REAL	32			0		read-write	
Constant a2 of the polynomial (2nd order coefficient)									
Name	Index:Sub	Type	Bit	Min	Max	Default Data	Unit	Access	PDO

	b		Si-ze	Da-ta	Da-ta		t		Map-ping
Con-stant a3	0x20-38:7	RE-AL	3-2			0		read-write	
Constant a3 of the polynomial (3rd order coefficient)									
Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Mi-n Da-ta	Ma-x Da-ta	Defau-lt Data	U-ni-t	Acc-ess	PD-O Map-ping
Con-stant a4	0x20-38:8	RE-AL	3-2			0		read-write	
Constant a4 of the polynomial (4th order coefficient)									
Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Mi-n Da-ta	Ma-x Da-ta	Defau-lt Data	U-ni-t	Acc-ess	PD-O Map-ping
Con-stant a5	0x20-38:9	RE-AL	3-2			0		read-write	
Constant a5 of the polynomial (5th order coefficient)									
Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Mi-n Da-ta	Ma-x Da-ta	Defau-lt Data	U-ni-t	Acc-ess	PD-O Map-ping

Cutoff frequency	0x2038:10	UINT	16	1	1000	500	Hz	read-write	
Cutoff frequency (Hz) for the LP filter for external scaled analog measurements.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Upper error threshold	0x2038:11	REAL	32			1199570688		read-write	
Upper error threshold for the external scaled measurements. When the scaled measurement value exceeds this threshold, a fault is generated. It is also used to generate a warning when the scaled measurement value is within 5% of the error threshold. The default value corresponds to the equivalent int representation of the float value 65535.0									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Lower error threshold	0x2038:12	REAL	32			0		read-write	
Lower error threshold for the external scaled measurements. When the scaled measurement value goes lower than this threshold, a fault is generated. It is also used to generate a warning when the scaled measurement value is within 5% of the error threshold.									

0x203F Error report

Provides further information about the error or warning reported by the `0x603F Error code` object.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Description	0x203F:1	STRING (8)	64			0		read-only (default)	Transmit or Receive PDO

The error report object display information about a fault or warning that has occured.
When a fault on the drive is reported, the drive switches into Fault reaction active state, which issues a quick-stop, brings the drive to a halt, and then disables the power stage. Thus, only one error is reported at a time.
When a warning is reported, the state of the drive remains unchanged. A subsequent fault overwrites an existing warning.

Error Code	Description	Explanation	Remedy
NoFault	No fault	Default value of the object, which means no fault is to be reported.	This error code indicates that there's no error reported, thus the drive is not in fault state. If the drive is shown in fault state and this error

Error Code	Description	Explanation	Remedy
			code is reported, this may be caused by a fault in the master/slave communication for example in the VM version OBLAC Drives. Rebooting the VM might solve the problem in this case. Another reason could be that the fault register could not be read. Otherwise please contact technical support of Synapticon.
Protection faults of user-defined parameters		Errors triggered by user-defined thresholds (for more details see object 0x2006: Protection)	NOTE: These errors can also cause subsequent faults or be caused by other faults.
PuOcA	Protection User-defined Over-cur-	The current	As the threshold is

Error Code	Description	Explanation	Remedy
	rent Phase A	sensor at drive phase A/B/C sensed a current exceeding the user-defined Over-current threshold (0x2006:3 Protection Over-current setpoint). If the user-defined threshold is higher than the hardware protection threshold, this software protection can be considered unfunctional, as long as the hardware protection is available.	user-defined, please first check the object 0x2006:3 Over-current setpoint to make sure that the threshold is set appropriately. Regarding an over-current, please refer to the remedy for over-current.
PuOcB	Protection User-defined Over-current Phase B		
PuOcC	Protection User-defined Over-current Phase C		
PuUv	Protection User-defined Under-voltage	Indicates that the DC bus voltage is measured below the user-defined threshold (0x2006:1 Protection Under-voltage setpoint).	As the threshold is user-defined, please first check the object 0x2006:1 Under-voltage setpoint. Then please refer to the remedy for

Error Code	Description	Explanation	Remedy
			under-voltage.
PuOv	Protection User-defined Over-voltage	Indicates that the DC bus voltage is measured exceeding the user-defined threshold (0x2006:2 Protection Over-voltage setpoint).	As the threshold is user-defined, please first check the object 0x2006:2 Over-voltage setpoint. Then please refer to the remedy for over-voltage.
Hardware Protection Faults		Errors triggered by hardware built-in protection logic.	
PhUdef	Protection Hardware Undefined	Indicates that the hardware protection reports an error not defined in the CiA 402 standard.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
PhFault	Protection Hardware Fault	A generic hardware fault was detected.	Please note down the information regarding the

Error Code	Description	Explanation	Remedy
			occurence of the error and reach Synapticon technical support for further investigation.
PhWtdg	Protection Hardware Watchdog	The watchdog is a service that checks periodically with a timer to guarantee that the drive control logic is working in the proper frequency. This error indicates that the watchdog tick is not received on time by the hardware protection, thus the drive is not working properly.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
PhDeadTA	Protection Hardware Deadtime Phase A	Deadtime is a common protection strategy in PWM control for avoiding a bridge	Please note down the information regarding the occurence of the error and reach

Error Code	Description	Explanation	Remedy
		shoot-through, a short-circuit accident caused by transistors on both sides being open at the same time. This error indicates that the effective deadtime is detected below the manufacturer-defined threshold.	Synapticon technical support for further investigation.
PhDeadTB	Protection Hardware Deadtime Phase B		
PhDeadTC	Protection Hardware Deadtime Phase C		
PhDeadTD	Protection Hardware Deadtime Phase D		
PhOvUvOt	Protection Hardware Overvoltage / Undervoltage / Overtemperature	The drive temperature which was measured close to the power MOSFETs and the DC bus voltage are checked by the same hardware protection circuit against manufacturer-defined thresholds. This error indicates that one or more of the three errors is/are triggered.	Please refer to the remedy for overvoltage, undervoltage and overtemperature.

Error Code	Description	Explanation	Remedy
PhOc	Protection Hardware Overcurrent	Indicate that the phase current or the DC bus current measured is exceeding the manufacturer-defined threshold (referring to the hardware document, link), which triggers the hardware protection circuit.	Please refer to the remedy for overcurrent.
Safety Faults			
SfeDiIvd	Safety Digital Input Invalid	Invalid digital input signals received by the safety input, for example one signal of the dual-channel input is read at a different interval than the other signal. The signals must have the same state within 100	The cause of the fault usually directs to misuse of the I/Os or faults in signal tansfer. Please refer to the documentation about safety functions of your hardware first, to check the use of the safety I/Os and of

Error Code	Description	Explanation	Remedy
		ms.	the safety functions. Check the measured inputs in 0x2611 Safety digital input diagnostics and locate the signal leading to the fault. Check and fix the wiring accordingly with assistance of tools for example with multimeter measurement.
SfeFault	Safety Fault	A safety-based hardware fault detected.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
BiSS Sensor Faults			
BisErBit	BiSS Error Bit active.	One of the bits in the BiSS frame is	Please check the 2

Error Code	Description	Explanation	Remedy
		used to signal error. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the error bit being active.	cases often seen: 1. Misconfiguration of Status bits active value. The status bits (error bit and warning bit) are by default read as active low, i.e. the error bit being 0 indicates the error being active. Please check the documentation of the encoder and correctly configure the reading. 2. The encoder is indeed rasing an error/ warning. Please check the documentation of the encoder, consult the encoder manufacturer for further investigation.
BisWnBit	BiSS Warning Bit active.	One of the bits in the BiSS frame is used to signal warning. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the warning bit being active.	

Error Code	Description	Explanation	Remedy
BisAcBit	BiSS Acknowledge Bit not received.	The BiSS slave is not acknowledging the request to communicate sent by the drive. This usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	Please firstly make sure that the encoder is connected to the port which is being configured. Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
BisSrtBt	BiSS Start Bit not received.	The BiSS slave acknowledged the request to communicate, but it didn't sent the start bit to start the communication. This usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or	

Error Code	Description	Explanation	Remedy
		not well connected.	
BisSloLo	BiSS data (SLO) is permanently Low.	The SLO line should be set to high by the BiSS slave when it is ready to establish communication. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	
BisFrame	BiSS Frame configuration error.	The sensor configuration causes an issue related to the BiSS frame. This error is currently raised only if the BiSS frame size is higher than 64 bits.	Please check the sensor configuration according to the documentation of the encoder manufacturer. The most relevant objects to check: 1. Resolution, unit

Error Code	Description	Explanation	Remedy
			Inc/ Revolution 2. Multiturn resolution, unit number of bits 3. Number of filling bits, unit number of bits
BisCrc	BiSS CRC error.	The BiSS frame contains a CRC6 of some of the data that is being sent. This error is raised when the CRC sent by the BiSS slave is different than the CRC computed by the drive. This usually means that the drive can't read the frame properly.	Regarding a CRC error, there are several causes often seen: 1. Misconfiguration of the sensor reading leading to the bits being read wrong. In this case the fault cannot be reset. Please check the sensor configuration according to the documentation of the encoder manufacturer. Common parameters wrongly configured are the singleturn

Error Code	Description	Explanation	Remedy
			resolution and multiturn resolution. 2. Misconfiguration of the object CRC polynom, thus the CRC is calculated wrong. In this case the fault cannot be reset. Please check the polynom in the encoder documentation. The value by default: 48 represents x^0 + x^1 + x^6, which usually doesn't need to be modified. 3. Inconsistency in signal transfer. In this case the fault can sometimes be reset, but would occur under certain circumstances. Please check the signals received by the drive with

Error Code	Description	Explanation	Remedy
			an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
BisRegEr	BiSS register communication error	Some BiSS encoders require the firmware to read/write some of their registers to	Try to reconfigure the encoder that raised this error. To do so, there are two possibilities: -

Error Code	Description	Explanation	Remedy
		be initialized properly. This error is triggered when any of these BiSS register transaction fail.	Set its function to Disabled and then back to the function it had. - Remove the encoder from the sensor port list (0x2100) and add it again.
SSI Sensor Faults			
SsiFrame	SSI Frame configuration error.	The sensor configuration causes an issue related to the SSI frame. This error is currently raised only if the SSI frame size is higher than 256 bits.	Please check the sensor configuration, especially the object Frame size according to the documentation of the encoder manufacturer.
SsiDtaLo	SSI Data is permanently Low.	The data line should be set to high by the SSI slave when it is ready to establish communication. This error usually	Please firstly make sure that the encoder is connected to the port which is being configured.

Error Code	Description	Explanation	Remedy
		happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
SsiChksm	SSI Checksum error.	Some SSI slaves include a ckeck-sum of the data in the SSI frame. Currently our drives can be configured to read a parity bit from compatible SSI encoders. This error means the parity bit com-	Regarding a check-sum error, there are several causes often seen: 1. Misconfiguration of the sensor leading to the bits being read wrong or the checksum being calculated

Error Code	Description	Explanation	Remedy
		puted by the firmware is different than the one sent by the SSI slave.	wrong. In this case the fault cannot be reset. Please check the sensor configuration according to the documentation of the encoder manufacturer. 2. Inconsistency in signal transfer. In this case the fault can sometimes be reset, but would occur under certain circumstances. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is

Error Code	Description	Explanation	Remedy
			unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
REM16MT Sensor Faults			
R16WkMgF	REM16 Weak Magnetic Field.	Weak magnetic field. Mounting may need to be improved.	
R16Cont	REM16 multiturn Counter error.	Multiturn counter error	
R16CorDc	REM16 singleturn CORDIC error	Singleturn CORDIC error	

Error Code	Description	Explanation	Remedy
R16SpdOv	REM16 multiturn Speed Overflow.	Multiturn speed overflow	
R16FtCfg	REM16 Filter Configuration error.	Filter configuration error	
R16FtSOF	REM16 Filter Speed Overflow error.	Filter speed overflow error	
R16UnCmd	REM16 Unknown Command.	Unknown command	
R16Chksm	REM16 Checksum error.	Checksum error	
REM14 Sensor Faults			
R14Chksm	REM14 Checksum error.	Checksum error	
QEI Sensor Faults			
QeiLsTck	QEI Lost Ticks.	The number of ticks counted in two successive index pulses is different from the sensor resolution	There are several causes often seen: 1. The configured resolution is

Error Code	Description	Explanation	Remedy
		configured by user.	wrong. Please check the Resolution configured according to the encoder datasheet, and please pay attention to the conversion between units (Resolution = Cycles per Revolution * 4). 2. There's a loss of pulses in encoder reading, which can be due to: 2a. The encoder being improperly mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and consult the encoder manufacturer for

Error Code	Description	Explanation	Remedy
			advices. 2b. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable

Error Code	Description	Explanation	Remedy
			noise. 2c. The pulses are read with an exceeding frequency. This case rarely happens, but can be possible if the encoder resolution and/or the spin speed being unusually high.
QeiNoIdx	QEI No Index.	For an incremental encoder configured to be with 3 channels (usually known as ABI/ ABZ), the drive is expecting to receive an Index pulse which provides an index for the absolute singleturn position. This error indicates that the Index pulse couldn't be detected, while	Please first check whether your encoder is indeed with the Index pulse, i.e. is an ABI/ ABZ encoder. If it is an encoder without index, please choose 2-channel (no index) for Number of channels. Else, most probable causes are: 1. The encoder being improperly

Error Code	Description	Explanation	Remedy
		the drive moved the motor in order to detect it.	mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and consult the encoder manufacturer for advices. 2. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging.

Error Code	Description	Explanation	Remedy
			Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
QeiOpnWr	QEI Open Wire.	Indicate that one of the differential A or B lines is disconnected. Only available on hardware that supports it.	Please check the wiring with for example the diode mode of a multimeter, making sure that the wires are stably transfering the signals.
Hall Sensor Faults			
HallSeq	Hall Sequence error.	The hall sensor outputs have an invalid value (111 or 000), or the sequence between the cur-	Please check the wiring and make sure that the signals are transfered with good con-

Error Code	Description	Explanation	Remedy
		rent state and the last state is invalid.	sistency. If the HALL sensor is used together with an incremental encoder for commutation, it can be that the HALL sensor A/B/C pins are connected in a wrong order. Please verify the cabling, and verify the order by running offset detection with only the HALL sensor configured for commutation. If the order is correct, the result of the offset detection should be around 0, and 0x2003:5 Motor phases inverted should show 0 (No).
A-Format sensor faults			

Error Code	Description	Explanation	Remedy
AftFrame	A-Format Frame error.	The fixed data in the recieved A-Format frame (like sync code) is wrong. This usually means that the A-Format transmission speed is wrong.	Please check the transmission speed configuration according to the encoder datasheet.
AftTmout	A-Format Timeout.	The A-Format encoder didn't start communication after sending a request.	Please check the wiring and make sure that the signals are transfered with good consistency. Use a oscilloscope to monitor the signal transfered for further investigation.
AftCrc	A-Format CRC error.	The A-Format frame contains a CRC of some of the data that is being sent. This error is raised when the CRC sent by the encoder is dif-	Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influ-

Error Code	Description	Explanation	Remedy
		ferent than the CRC computed by the drive. This usually means that the drive can't read the frame properly, which usually happens because of noise in the communication link.	enced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
AftBatt	A-Format BATT status.	The encoder is raising Battery Alarm. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Battery Alarm. The usual case is that the battery voltage is low, the battery is wrongly wired, or there's no bat-

Error Code	Description	Explanation	Remedy
			tery connected.
AftMTErr	A-Format MTERR status.	The encoder is raising MT Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the MT Error.
AftOvFlw	A-Format OvFlow status.	The encoder is raising Over Flow Alarm. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Over Flow Alarm.
AftOvSpd	A-Format OVSPD status.	The encoder is raising Over Speed Alarm. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Over Speed Alarm. The usual case is that the rotation speed exceeds the speed limit of the encoder.
AftMemEr	A-Format MEMERR status.	The encoder is raising Memory	Please check the encoder doc-

Error Code	Description	Explanation	Remedy
		Error. Check your A-Format encoder datasheet for more information.	umentation for better understanding the meaning of the Memory Error.
AftSTErr	A-Format STERR status.	The encoder is raising ST Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the ST Error.
AftPSErr	A-Format PSERR status.	The encoder is raising PS Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the PS Error.
AftBusyF	A-Format BUSY status.	The encoder is raising BUSY Flag. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the BUSY Flag.
AftMemBs	A-Format MemBusy status.	The encoder is raising MEMBUSY Flag. Check your A-Format encoder	Please check the encoder documentation for better understanding the meaning of the

Error Code	Description	Explanation	Remedy
		datasheet for more information.	MEMBUSY Flag.
AftOvTmp	A-Format OvTemp status.	The encoder is raising Temperature Warning. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Temperature Warning. In case of over-temperature please refer to the remedies for over-temperature.
AftIncEr	A-Format IncErr status.	The encoder is raising Incremental Signal Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Incremental Signal Error.
Generic Sensor Faults			
SnsrAngl	Sensor Angle error.	1. The electrical angle was not completeting a whole revolution	Please firstly check whether the motor rotates dur-

Error Code	Description	Explanation	Remedy
		during offset detection. 2. The calculated commutation angle is out of bounds. A singleturn related configuration parameter (resolution, singleturn) with a wrong value normally causes this error. 3. The motor is not moving during the QEI index detection.	ing offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. Please check the encoder configuration, and make sure that while rotating the motor for one full turn, the encoder value is correctly responding to the mechanical position. You can manually rotate the motor for investigation. An alternative way is with OBLAC opening another page and ploting the position feedback of the encoder used for commutation while running offset detection.

Error Code	Description	Explanation	Remedy
SnsrCfg	Sensor Configuration error.	1. Some encoder parameter was set with an invalid value in the Object dictionary. 2. An action that required a particular encoder function (position, velocity or commutation) was triggered, but there is no encoder configured for that function. 3. There was an issue when trying to initialize or configure an encoder.	Please check the configuration of the sensors, especially the parameters changed lately. Make sure that all functions required (position, velocity, commutation) are configured with the sensors.
HwRsrcEr	Hardware Resource Error.	The software configuration is invalid for the hardware. For example, a sensor configured is with a sensor type not supported by that port.	Please check the sensor configurations and the digital IO settings, making sure that the setting is supported by your hardware according to the doc-

Error Code	Description	Explanation	Remedy
			umentation.
IvldGpio	Invalid Gpio configuration.	The GPIO related configuration chosen is not available. This includes voltage level, direction of the GPIO, or GPIO trigger.	Please check the configuration of the GPIO according to the documentation.
SnsrBatt	Sensor battery error	This error is raised when an encoder battery was expected, but couldn't be detected. This error is raised only by multiturn encoders that need a battery, like Circulo's Internal encoder 1 when it's configured as a multiturn encoder.	Please check the documentation to make sure the battery is connected properly and power cycle. In case there is no battery or if the battery is not needed, select "0" or "No" to the "Multiturn resolution" parameter of this encoder.
SkpCycls	Serial encoder service is skipping cycles	This error is raised when a serial encoder service is looping	To try to solve either the warning or the error, we should try to spend

Error Code	Description	Explanation	Remedy
		slower than 16 kHz, which means that the cycles are taking longer than 62.5 us. This error should not be confused with the warning of the same name, which just means that an encoder that supports 32 kHz sampling (BiSS or SSI) is looping at 16 kHz instead.	less time reading the encoder. To do so we can first try to increase the clock frequency, and then try reducing the encoder timeout (check the encoder datasheet to determine which is the minimum value).
SwLimOut	Software Limit Out.	Target position is outside of the 0x607D: Software position limit defined by the user.	Please check object 0x607D: Software position limit and make sure that they are configured as expected. Accordingly send your position target.
Brake Faults			

Error Code	Description	Explanation	Remedy
BrkNtRls	Brake release failed	Brake release procedure for pin brake failed, i.e. the configured displacement threshold (0x2004:8 Minimum displacement (pin brake)) is not reached with the configrued torque (0x2004:9 Maximum torque (pin brake)).	Please firstly check whether the motor functions correct, for example by manually releasing the brake and running offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. While the motor functions correct, please further tune the two configuration parameters, by increasing the torque or decreasing the minimum displacement, until the brake can be released stably.
BkHiPull	Pull brake voltage too high	Pull brake voltage (0x2004:1 Pull voltage) higher than measured	Please measure the main power supply voltage sup-

Error Code	Description	Explanation	Remedy
		DC bus voltage (0x6079: DC link circuit voltage).	plied to the drive, and check whether the result is close to 0x6079: DC link circuit voltage. Pay attention that the actual voltage supplied to the drive would be slightly lower than the voltage output by the power supply, because of attenuation. Check the configuration of the pull/hold voltage, which should be lower than the DC Bus voltage.
BkHiHold	Hold brake voltage too high	Hold brake voltage (0x2004:2 Hold voltage) higher than measured DC bus voltage (0x6079: DC link circuit voltage).	
Open Terminal Faults			
OpnTermA	Open Terminal A	Drive terminal A is opened. This error can happen if motor cable is not connected to terminal A of the	Please check the motor wiring, making sure that the motor phase cable is connected to the terminal A/B/C of

Error Code	Description	Explanation	Remedy
		Drive. It can also happen if the Drive is damaged internally.	the Drive.
OpnTermB	Open Terminal B	Drive terminal B is opened. This error can happen if motor cable is not connected to terminal B of the Drive. It can also happen if the Drive is damaged internally.	
OpnTermC	Open Terminal C	Drive terminal C is opened. This error can happen if motor cable is not connected to terminal C of the Drive. It can also happen if the Drive is damaged internally..	
Open FET Faults			

Error Code	Description	Explanation	Remedy
OpnFetAH	Open high FET at phase A	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg A of the inverter, and it is the high-side FET of this leg.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
OpnFetBH	Open high FET at phase B	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg B of the inverter, and it is the high-side FET of this leg.	
OpnFetCH	Open high FET at phase C	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg C of the inverter, and it	

Error Code	Description	Explanation	Remedy
		is the high-side FET of this leg.	
OpnFetAL	Open low FET at phase A	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg A of the inverter, and it is the low-side FET of this leg.	
OpnFetBL	Open low FET at phase B	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg B of the inverter, and it is the low-side FET of this leg.	
OpnFetCL	Open low FET at phase C	One of the power electronic switches of Drive is damaged (not conducting). The	

Error Code	Description	Explanation	Remedy
		damaged FET is located in leg C of the inverter, and it is the low-side FET of this leg.	
Parameter Faults			
IvldOfst	Invalid offset	The commutation offset in 0x2001: Commutation angle offset is not valid. This error comes directly from reading object 0x2009:1 Commutation offset: State.	Please run offset detection before operations. Please pay attention that, the change in some fundamental configuration parameters (sensor configuration, motor configuration) can make the existing offset detection result invalid. Please redo offset detection in that case.
IvldOpmd	Invalid operational mode	Requested opmode is not supported. Set a valid value in 0x6060:	Please check our documentation for the available operation modes.

Error Code	Description	Explanation	Remedy
		Modes of operation.	Please check 0x6060: Modes of operation, and make sure that the master controller is sending a correct command.
ZeroMxI	Max. current is zero or negative	User-defined limit on motor max current (0x6073: Max current) is 0.	Please configure the 0x6073: Max current to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxT	Max. torque is zero or negative	User-defined limit on motor max current (0x6072: Max torque) is 0.	Please configure the 0x6072: Max torque to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxMS	Max. motor speed is zero or negative	The value of 0x6080: Max motor speed is 0, so no motion is allowed.	Pleaes configure 0x6080: Max motor speed to a valid value larger than 0, according

Error Code	Description	Explanation	Remedy
			to your motor data-sheet or your application need.
IvldPara	Invalid Parameter	The value of a parameter is invalid.	Please check the values changed lately according to the doc-umentation.
SiVelPfx	Invalid velocity scaling factor	The velocity scal-ing prefix defined by 0x60A9: SI unit velocity is outside of the supported range.	Please check OBLAC Drives, which directly provides available options for the con-figuration of 0x60A9: SI unit velocity, and the documentation for user defined velo-city units. Make sure that a valid data is write to the object.
MxPwrLmt	Maximum mech-anical output power of the motor is reached.	This warning appears when the limit has been reached that has been configured in	When this waring appears often in your setup and your application does not perform

Error Code	Description	Explanation	Remedy
		object 0x200B: Max power. The warning will disappear when the absolute value of reference torque has become less than this limit.	as intended please consider the following remedies: - set the limit higher if your application allows it - reconsider the size of your power-supply - choose a more powerful motor
Communication Faults			
CyclicHb	Cyclic heartbeat	A communication keep-alive heartbeat was dropped. Check that the PDO messages are being sent by the master cyclically, and within 100 ms of each other.	If the inactivity of communication is known/ intended, the occurrence of the error is expected. Do [Fault reset] to clear the fault after the communication is reestablished. Otherwise, the occurrence of the error indicates that

Error Code	Description	Explanation	Remedy
			there exists unknown issues in the communication chain. Possible causes include: - communication delay in master program, check the running status of the master machine. - disturbance in communication signal transfer, make sure that a communication cable with shielding is used or try with the cable provided by Synapticon, check the stability of cabling.
OsCmdCol	OS command collision	A new OS command was sent, but the OS command response of a previous OS command was not read after it com-	In order to send another OS command, please read the response of the previous OS command in the object 0x1023:3.

Error Code	Description	Explanation	Remedy
		pleted.	Whenever the first byte becomes 0, 1, 2 or 3, a new OS command can be sent.
I2t Protection Errors			
I2tActv	I2t protection is active	I2t protection is active, which limits your maximum torque to the nominal rated values of the motor. See 0x200A: I2t for configuration parameters.	Please check 0x200A: I2t and see whether it's configured as expected. Regarding the I2t being triggered, that indicates that the drive is protecting the motor from potential overheating. Please read our documentation of I2t for more information.
Homing errors			

Error Code	Description	Explanation	Remedy
HmInvMth	Unsupported or invalid homing method	A Homing method was selected that is not supported by SOMANET.	Please firstly check object 0x6098: Homing method and see whether the object is written as expected. Please refer to our documentation Homing modes, and check what homing methods are supported.
Internal Errors			
Intern01	Internal fault 1	Internal Error Code. Please contact support.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
Intern02	Internal fault 2	Internal Error Code. Please contact support.	
Intern03	Internal fault 3	Internal Error Code. Please contact support.	

Error Code	Description	Explanation	Remedy
Filter cutoff frequency bounds			
VeFiFcLo	Velocity filter cut-off frequency too low	0x2021:2 Velocity feedback filter Cutoff frequency is lower than the minimum limit.	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
VeFiFcHi	Velocity filter cut-off frequency too high	0x2021:2 Velocity feedback filter Cutoff frequency is higher than the maximum limit.	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.

Error Code	Description	Explanation	Remedy
PoFiFcLo	Position filter cut-off frequency too low	0x2022:2 Position feedback filter Cutoff frequency is lower than the minimum limit.	Please check 0x2022:2 Position feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
PoFiFcHi	Position filter cut-off frequency too high	0x2022:2 Position feedback filter Cutoff frequency is higher than the maximum limit.	Please check 0x2022:2 Position feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
VffFcLo	Velocity feed forward filter cut-off	0x2015:2 Velocity feed forward	Please check 0x2015:2 Velocity

Error Code	Description	Explanation	Remedy
	frequency too low	Cutoff frequency is lower than the minimum limit.	feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
VffFcHi	Velocity feed forward filter cut-off frequency too high	0x2015:2 Velocity feed forward Cutoff frequency is higher than the maximum limit.	Please check 0x2015:2 Velocity feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
NhFiFcHi	Notch filter center frequency too high	0x2023:2 Notch filter Center frequency is higher than the maximum	Please check 0x2023 Notch filter, and make sure that the center fre-

Error Code	Description	Explanation	Remedy
		limit.	quency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
NhFiPara	Invalid Notch filter parameters	Invalid configuration was detected for the notch filter.	Please check 0x2023 Notch filter, and make sure that the rejection band is set smaller than the center frequency.
Distributed Clock			
SynDifHi	Sync difference too high	Sync time variance of the EtherCAT distributed clock is too high (value will be ignored).This may happen if the distributed clock is poorly configured or the signal isn't consistent.	Please check the distrubuted clock (DC) setting of your EtherCAT master.

Error Code	Description	Explanation	Remedy
Temperature Sensor Faults			
TmpCore		This fault is triggered when the temperature of the core becomes too high and crosses the error threshold (> 100 °C). This happens usually because of sustained load on the processor, high operating frequency and even high ambient temperature. High core temperature can lead to equipment damage. It can also result in irregular clock timing.	Throttling the core is usally the remedy but as this is not possible in the case of real-time critical applications like ours, the solution would be to disable the drive until the reason for the error is resolved or use external cooling.
TmpDrive		This fault is triggered when the temperature of the drive becomes	Disable the drive until the reason for the error is resolved or use external cooling to

Error Code	Description	Explanation	Remedy
		too high and crosses the error threshold (> 100 °C). This happens usually because of sustained load on the processor, current leakage from the on-board componenets, high operating frequency and even ambient temperature. High drive temperature can lead to equipment damage.	bring down the over-all temperature.

0x2040 Input counter

Unsigned 32bit integer, resets after 4,294,967,295 (2^32 - 1) µs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Input counter	0x2040:0	UDINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x20E1 High resolution data

Prototype - This object will change in future releases!

High-resolution data contains all position, velocity and phase current measurements of the last millisecond. It is updated once per millisecond with the new data.

There are four sub-objects, which, when concatenated, reflect the full chunk of data. The timestamp corresponds to the measurements marked "1". Time step between position/velocity values is the motion control loop period, time step between phase currents is the torque control loop period. The data is encoded as follows:

Bytes	Value
0-3	Timestamp (32-bit unsigned int)
4-7	Position 1 (32-bit signed int)
8-11	Position 2 (32-bit signed int)
12-15	Position 3 (32-bit signed int)
16-19	Position 4 (32-bit signed int)
20-23	Velocity 1 (32-bit signed int)
24-27	Velocity 2 (32-bit signed int)
28-31	Velocity 3 (32-bit signed int)
32-35	Velocity 4 (32-bit signed int)
36-37	Phase Current I_b 1 (16-bit signed int)
38-39	Phase Current I_c 1 (16-bit signed int)
40-41	Phase Current I_b 2 (16-bit signed int)

Bytes	Value
42-43	Phase Current I_c 2 (16-bit signed int)
...	...
96-97	Phase Current I_b 16 (16-bit signed int)
98-99	Phase Current I_c 16 (16-bit signed int)

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 1	0x20E1:1	OCTET_STRING (25)	200			0		read-only (default)	Transmit PDO (Inputs)
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 2	0x20E1:2	OCTET_STRING	200			0		read-only (default)	Transmit PDO (Inpu-

		(25)							ts)
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 3	0x20-E1:3	OCTET_STRING (25)	20-0			0		read-only (default)	Transmit PDO (Inputs)
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 4	0x20-E1:4	OCTET_STRING (25)	20-0			0		read-only (default)	Transmit PDO (Inputs)

0x20F0 Timestamp

Position signal measurement timestamp. Unsigned 32bit integer, resets after 42,949,672 ((2^32 - 1) / 100) µs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Time-	0x20-	UD-	32	0	42949-	0	µs	read-	Tran-

stam-p	F0:0	INT			67295			only (def-ault)	smit PDO (Inp-uts)

0x20F2 Assigned name

Allows a unique name to be assigned to this drive. This is only for reference; it is not used by the firmware.

Nam-e	Index-:Sub	Type	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Assi-gned nam-e	0x20-F2:0	STR-ING (50)	40-0					read-write	

0x20F3 DC synchronization

Indicates the status of the selected network synchronization mechanism.

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
DC pul-se tim-e	0x20F-3:1	UDI-NT	32			0		read-write	
The time when the last DC pulse happened in clock ticks (10 ns).									

0x2110 Encoder 1 configuration

All configuration options for Encoder 1

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Sensor port	0x2110:1	USINT	8	No sensor port: 0 Sensor port 1: 1 Sensor port 2: 2 Sensor port 3: 3 Sensor port 4: 4 Sensor port 5: 5		readwrite	

All encoders

The sensor port on which this encoder is connected.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping

			e				ng
Type	0x21-10:2	US-IN-T	8	None 0 Hall sensor 1 Incremental encoder 2 Nikon A-format 3 BiSS 4 REM 16MT 6 SSI 7 SinCos module 9		rea-dwri-te	

All encoders

Select the type of encoder you want to use.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Resolu-	0x21-10:3	U-DI-	3-2	1		1	Inc/Revolu-	rea-dwri-	

tion		NT					tion	te	

BiSS/SSI

Number of pulses in one revolution.
Most of BiSS/SSI encoders have a binary resolution, which means:
Resolution = 2^(subindex 12 *Singleturn bits*).

Incremental encoder

Total number of quadrature edges per revolution.
Resolution = Cycles per Revolution * 4

Hall sensor/Nikon A-Format/REM16MT/SinCos module

For these encoders this parameter is configured automatically.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Zero velocity threshold	0x2110:4	UDINT	32	0	1000000	4000	µs	readwrite	

Incremental encoder

This parameter is only used for the velocity calculation of incremental encoders.
If no new tick is received from the encoder during the time specified in this parameter, the velocity of the encoder will be set to 0.
The value will automatically be rounded down to a multiple of 62.5 us.

Name	Index:Sub	Type	Bit S-	Options	Unit	Access	PDO Ma-

			iz-e				ppi-ng
Polar-ity	0x21-10:5	B-O-OL	1	Normal polarity: 0 Inverted polarity: 1		rea-dwri-te	

All encoders

Invert the direction of movement of the encoder. This is the equivalent of physically flipping the encoder, so the commutation angle offset will not be valid after changing this parameter. If you have an axis with two encoders (one for commutation and one for position and velocity), the direction of each encoder must match for control stability. This polarity parameter can be used for flipping the direction of the position or velocity encoder relative to the sensor used for commutation.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
Sing-leturn offset	0x21-10:6	U-DI-NT	3-2			0	Inc	rea-dwri-te	

Absolute encoders (BiSS/SSI/Nikon A-format/REM 16MT)

This value is substracted from the singleturn value received from the encoder. It is the equivalent of physically moving the encoder, so the commutation angle offset will not be valid after changing this parameter. This parameter has to be lower than the resolution of the encoder.
The singleturn offset is applied after the subindex 5 *Polarity* has been applied.

Nam-	Inde-	Ty-	B-	Options	Unit	Acc-	PD-

e	x:Sub	pe	it Size			ess	O Mapping
Access signal type	0x2110:7	BOOL	1	Differential RS422	0	readwrite	
				Single-ended TTL	1		

Incremental encoder

Select if the encoder signal is differential (RS422) or single-ended (TTL).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Clock frequency	0x2110:8	UINT	16	1000	16000	1000	kHz	readwrite	

BiSS/SSI

Frequency of the SSI clock line or BiSS MA line.
Refer to your encoder datasheet to check what are the minimum and maximum values for this parameter.
The drive can only clock at some discrete frequencies. If the user enters a frequency that is not within these discrete frequencies, it will be set to the next possible lower frequency.
To calculate what's the real frequency that will be used, you can round up the result of

25000 / Clock frequency. Then divide 25000 by the rounded value obtained before.
For example, when setting 4000 KHz the real frequency will be 3571.43 kHz.
The maximum value that can be set is 5000 kHz.

Nikon A-format

Select the transmission speed of the encoder. The only possible values are 2500, 4000, 6670, 8000 and 16000 kbps.
See encoder datasheet for more information.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Frame size	0x2110:9	USINT	8	1		1		readwrite	

BiSS

The BiSS service expects an encoder that has the following frame structure:
- SLO line high when encoder is ready to start communicating.
- Ack bit (which can be extended to multiple bits by the encoder).
- Start bit when the encoder is ready to transmit the data.
- CDS bit for register communication.
- Payload (any data can be sent here, but the only data that can be processed is Multiturn and singleturn position).
- Error and warning bits, in this order.
- Variable length CRC.
- Timeout.

This parameter specifies the number of bits transmited as Payload.

SSI

This parameter specifies the number of bits transmited in one encoder frame.

Name	Index	Type	Bit	Min	Max	Default	Unit	Access	PDO

e	x:Sub	pe	it Size	Data	Data	ault Data		ess	O Mapping
Multi-turn bits	0x21-10:1-0	USINT	8			0		read-write	

BiSS/SSI/Nikon A-format

Number of multiturn bits to read from the encoder.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Multi-turn first bit position	0x21-10:1-1	USINT	8			0		read-write	

BiSS/SSI

Position of the first bit of multiturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*.
Usually this value is 0, since multiturn is the first element on the frame.
Only relevant if subindex 10 *Multiturn bits* is bigger than 0.

Name	Index:Sub	Type	Bit S-	Min Data	Max Data	Default Dat-	Unit	Access	PDO Ma-

			iz-e			a			ppi-ng
Sing-leturn bits	0x21-10:1-2	US-IN-T	8	1		1		rea-dwri-te	

BiSS/SSI

Number of singleturn bits to read from the encoder.

Nikon A-format

Number of singleturn bits to read from the encoder. The only possible values are 17, 20, 22 and 24.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
Sing-leturn first bit pos-ition	0x21-10:1-3	US-IN-T	8			0		rea-dwri-te	

BiSS/SSI

Position of the first bit of singleturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*. For most of the BiSS/SSI encoders, this value is equal to: subindex 10 *Multiturn bits* + subindex 11 *Multiturn first bit position*.

Nam-e	Inde-x:Su-	Ty-pe	B-it	Min Dat-	Max Dat-	Def-ault	Unit	Acc-ess	PD-O

	b		S-iz-e	a	a	Dat-a			Ma-ppi-ng
Time-out	0x21-10:1-4	US-IN-T	8			36	µs	rea-dwri-te	

BiSS/SSI

Timeout that will be sent by the drive at the end of each BiSS/SSI frame, which informs the encoder that the frame is over.
Refer to your encoder datasheet to check what is the minimum value for this parameter.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
CRC poly-nomi-al	0x21-10:1-5	US-IN-T	8			67		rea-dwri-te	

BiSS

Polynomial for the CRC calculation of the sensor data (SCD).
This value is obtained by writing the polynomial in binary representation.
For example: x^6 + x^3 + x^1 + x^0 is 0b1001011 = 0x4B = 75.
The default value of 67 (x^6 + x^1 + x^0 = 0b1000011 = 0x43) is the one used for most of the BiSS encoders.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-

			e						ng
Maximum tbusy	0x2110:16	USINT	8			20	µs	readwrite	

BiSS

This value defines the maximum time to wait for the Start bit once the Ack bit was already received. If the Start bit is not sent after this time, the error BisSrtBt is raised.
Check your encoder datasheet to find what's the typical or maximum value for this parameter.
It is advised to try to increase this number in case the error BisSrtBt appears, which is likely to happen because the encoder needed more time to process the data before sending the Start bit.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Status bits active value	0x2110:17	USINT	8	Status bits active low: 0 Status bits active high: 1		readwrite	

BiSS

Value of the status bits when active. The BiSS-C standard recommends that these bits are active low, so most encoders should work with the default value.

Name	Index:Sub	Type	B-	Options	Unit	Access	PD-

	b		it S-iz-e				O Ma-ppi-ng
Par-ity type	0x21-10:1-8	US-IN-T	8	No par-ity: 0 Odd parity: 1 Even parity: 2		rea-dwri-te	

SSI

Type of parity read from the encoder. If there is parity, it should be the last bit on the SSI frame.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
First clock delay	0x21-10:1-9	UI-NT	1-6			0	µs	rea-dwri-te	

SSI

Delay after first clock edge. This is compatible only with some encoders (will be specified in the encoder datasheet). If your encoder supports this function, it's normally used to be able to increase the maximum allowable clock frequency.

Nam-	Inde-	Ty-	B-	Options	Unit	Acc-	PD-

e	x:Sub	pe	it Size			ess	O Mapping
Data ordering	0x2110:20	BOOL	1	Natural Binary Code: 0 Reflected Binary Code: 1		readwrite	

SSI

Select if the ordering of position is done using Natural Binary Code or Reflected Binary Code (also known as Gray code).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Endianness	0x2110:21	BOOL	1	MSB first: 0 LSB first: 1		readwrite	

SSI

Order of the bits (Most Significant Bit or Least Significant Bit transmitted first). Usually MSB is transmitted first.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Index availability	0x2110:22	BOOL	1	Encoder without index: 0 Encoder with index: 1		readwrite	

Incremental encoder/SinCos module

Define if the connected encoder has an index or not.
For SinCos encoders the index is often called "reference".
Incremental encoders without index are often called AB encoders, while incremental encoders with index are often called ABI encoders.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Hall sensor port	0x2110:23	USINT	8	No sensor port: 0 Sensor port 1: 1		readwrite	

				<table><tr><td>Sensor port 2</td><td>2</td></tr><tr><td>Sensor port 3</td><td>3</td></tr><tr><td>Sensor port 4</td><td>4</td></tr><tr><td>Sensor port 5</td><td>5</td></tr></table>					

Incremental encoder

This parameter is used to define to which Sensor port the Hall sensor is connected in case of using it together with an incremental encoder in order to have an approximate absolute position while the incremental encoder index is not yet found.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Sinewave cycles per revolution	0x2110:24	UINT	16	1		1		readwrite	

SinCos module

Number of sinewaves cycles in one motor revolution.
This value is going to be used to calculate the singleturn resolution (subindex 3 *Resolution*).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Sinewave resolution	0x2110:25	USINT	8	8 13 16 12 32 11 64 10 128 9 256 8 512 7 1024 6 2048 5 4096 4 8192 3	Inc	readwrite	

SinCos module

Select desired number of pulses in one sinewave.
This value is going to be used to calculate the singleturn resolution (sunindex 3 *Resolution*).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping

			e				ng
Sinusoidal output voltage	0x21-10:2-6	USINT	8	up to 50 mVpp: 15 up to 60 mVpp: 14 up to 75 mVpp: 13 up to 0.1 Vpp: 12 up to 0.12 Vpp: 11 up to 0.14 Vpp: 10 up to 0.15 Vpp: 9 up to 0.2 Vpp: 8		read-write	

up to 0.28 Vpp	7
up to 0.4 Vpp	6
up to 0.5 Vpp	5
up to 0.6 Vpp	4
up to 0.75 Vpp	3
up to 1 Vpp	2
up to 1.2 Vpp	1
up to 1.5 Vpp	0

SinCos module

Differential voltage in the Sin/Cos outputs peak to peak.
It is often abreviated as Vpp.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Filter	0x2110:27	USINT	8	Disabled: 0 Filter strength 2 (weakest): 2 Filter strength 3: 3 Filter strength 4: 4 Filter strength 5: 5 Filter strength 6: 6 Filter strength 7: 7 Filter: 8		readwrite	

				strength 8				
				Filter strength 9 (stronge-st)	9			

REM 16MT

The filter with strength F of the sensor. The estimated step response time of the filter is Ts*2^F (Ts is the sampling time, ca. 70µs).

0x2111 Encoder 1 feedback

Feedback of the encoder configured as Encoder 1

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Raw position	0x2111:1	UDINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)
The position value directly read from the encoder.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Adjusted position	0x211-1:2	DINT	32			0		read-only (default)	Transmit PDO (Inputs)
The position value after applying the Singleturn offset and the Sensor polarity and the location of the index. This is an absolute value.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity	0x211-1:3	DINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x2112 Encoder 2 configuration

All configuration options for Encoder 2

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Sensor port	0x2112:1	USINT	8	No sensor port	0		readwrite	

				Sensor port 1: 1 Sensor port 2: 2 Sensor port 3: 3 Sensor port 4: 4 Sensor port 5: 5			

All encoders

The sensor port on which this encoder is connected.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Type	0x2112:2	USINT	8	None: 0 Hall sensor: 1 Incremental encoder: 2 Nikon A-: 3		readwrite	

				format					
				BiSS	4				
				REM 16MT	6				
				SSI	7				
				SinCos module	9				

All encoders

Select the type of encoder you want to use.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Resolution	0x21-12:3	UDINT	32	1		1	Inc/Revolution	readwrite	

BiSS/SSI

Number of pulses in one revolution.
Most of BiSS/SSI encoders have a binary resolution, which means:
Resolution = 2^(subindex 12 *Singleturn bits*).

Incremental encoder

Total number of quadrature edges per revolution.
Resolution = Cycles per Revolution * 4

Hall sensor/Nikon A-Format/REM16MT/SinCos module

For these encoders this parameter is configured automatically.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Zero velocity threshold	0x2112:4	UDINT	32	0	1000000	4000	µs	readwrite	

Incremental encoder

This parameter is only used for the velocity calculation of incremental encoders.
If no new tick is received from the encoder during the time specified in this parameter, the velocity of the encoder will be set to 0.
The value will automatically be rounded down to a multiple of 62.5 us.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Polarity	0x2112:5	BOOL	1	Normal polarity: 0 Inverted polarity: 1		readwrite	

All encoders

Invert the direction of movement of the encoder. This is the equivalent of physically flipping

the encoder, so the commutation angle offset will not be valid after changing this parameter. If you have an axis with two encoders (one for commutation and one for position and velocity), the direction of each encoder must match for control stability. This polarity parameter can be used for flipping the direction of the position or velocity encoder relative to the sensor used for commutation.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Singleturn offset	0x2112:6	UDINT	32			0	Inc	readwrite	

Absolute encoders (BiSS/SSI/Nikon A-format/REM 16MT)

This value is substracted from the singleturn value received from the encoder. It is the equivalent of physically moving the encoder, so the commutation angle offset will not be valid after changing this parameter. This parameter has to be lower than the resolution of the encoder.
The singleturn offset is applied after the subindex 5 *Polarity* has been applied.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Access signal type	0x2112:7	BOOL	1	Differential RS422: 0 Single-: 1		readwrite	

				ended TTL					

Incremental encoder

Select if the encoder signal is differential (RS422) or single-ended (TTL).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Clock frequency	0x2112:8	UINT	16	1000	16000	1000	kHz	readwrite	

BiSS/SSI

Frequency of the SSI clock line or BiSS MA line.
Refer to your encoder datasheet to check what are the minimum and maximum values for this parameter.
The drive can only clock at some discrete frequencies. If the user enters a frequency that is not within these discrete frequencies, it will be set to the next possible lower frequency.
To calculate what's the real frequency that will be used, you can round up the result of 25000 / Clock frequency. Then divide 25000 by the rounded value obtained before.
For example, when setting 4000 KHz the real frequency will be 3571.43 kHz.
The maximum value that can be set is 5000 kHz.

Nikon A-format

Select the transmission speed of the encoder. The only possible values are 2500, 4000, 6670, 8000 and 16000 kbps.
See encoder datasheet for more information.

Name	Index:Sub	Type	B-	Min Dat-	Max Dat-	Def-	Unit	Access	PD-

	b		it S-iz-e	a	a	ault Dat-a			O Ma-ppi-ng
Fram-e size	0x21-12:9	US-IN-T	8	1		1		rea-dwri-te	

BiSS

The BiSS service expects an encoder that has the following frame structure:
- SLO line high when encoder is ready to start communicating.
- Ack bit (which can be extended to multiple bits by the encoder).
- Start bit when the encoder is ready to transmit the data.
- CDS bit for register communication.
- Payload (any data can be sent here, but the only data that can be processed is Multiturn and singleturn position).
- Error and warning bits, in this order.
- Variable length CRC.
- Timeout.

This parameter specifies the number of bits transmited as Payload.

SSI

This parameter specifies the number of bits transmited in one encoder frame.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
Multi-turn bits	0x21-12:1-0	US-IN-T	8			0		rea-dwri-te	

BiSS/SSI/Nikon A-format

Number of multiturn bits to read from the encoder.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Multiturn first bit position	0x2112:1	USINT	8			0		readwrite	

BiSS/SSI

Position of the first bit of multiturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*.
Usually this value is 0, since multiturn is the first element on the frame.
Only relevant if subindex 10 *Multiturn bits* is bigger than 0.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Singleturn bits	0x2112:2	USINT	8	1		1		readwrite	

BiSS/SSI

Number of singleturn bits to read from the encoder.

Nikon A-format

Number of singleturn bits to read from the encoder. The only possible values are 17, 20, 22 and 24.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Singleturn first bit position	0x2112:13	USINT	8			0		readwrite	

BiSS/SSI

Position of the first bit of singleturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*. For most of the BiSS/SSI encoders, this value is equal to: subindex 10 *Multiturn bits* + subindex 11 *Multiturn first bit position*.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Timeout	0x2112:14	USINT	8			36	µs	readwrite	

BiSS/SSI

Timeout that will be sent by the drive at the end of each BiSS/SSI frame, which informs the encoder that the frame is over.

Refer to your encoder datasheet to check what is the minimum value for this parameter.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
CRC polynomial	0x2112:15	USINT	8			67		readwrite	

BiSS

Polynomial for the CRC calculation of the sensor data (SCD).
This value is obtained by writing the polynomial in binary representation.
For example: x^6 + x^3 + x^1 + x^0 is 0b1001011 = 0x4B = 75.
The default value of 67 (x^6 + x^1 + x^0 = 0b1000011 = 0x43) is the one used for most of the BiSS encoders.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Maximum tbusy	0x2112:16	USINT	8			20	µs	readwrite	

BiSS

This value defines the maximum time to wait for the Start bit once the Ack bit was already received. If the Start bit is not sent after this time, the error BisSrtBt is raised.
Check your encoder datasheet to find what's the typical or maximum value for this parameter.

It is advised to try to increase this number in case the error BisSrtBt appears, which is likely to happen because the encoder needed more time to process the data before sending the Start bit.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Status bits active value	0x2112:17	USINT	8	Status bits active low: 0 Status bits active high: 1		read-write	

BiSS

Value of the status bits when active. The BiSS-C standard recommends that these bits are active low, so most encoders should work with the default value.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Parity type	0x2112:18	USINT	8	No parity: 0 Odd parity: 1		read-write	

				Even parity	2				

SSI

Type of parity read from the encoder. If there is parity, it should be the last bit on the SSI frame.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
First clock delay	0x2112:19	UINT	16			0	µs	readwrite	

SSI

Delay after first clock edge. This is compatible only with some encoders (will be specified in the encoder datasheet). If your encoder supports this function, it's normally used to be able to increase the maximum allowable clock frequency.

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Data ordering	0x2112:20	BOOL	1	Natural Binary Code	0		readwrite	
				Reflec-	1			

				ted Binary Code			

SSI

Select if the ordering of position is done using Natural Binary Code or Reflected Binary Code (also known as Gray code).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Endianness	0x2112:21	BOOL	1	MSB first: 0 LSB first: 1		readwrite	

SSI

Order of the bits (Most Significant Bit or Least Significant Bit transmitted first). Usually MSB is transmitted first.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Index availabil-	0x2112:22	BOOL	1	Encoder without: 0		readwrite	

ity				index			
				Encoder with index	1		

Incremental encoder/SinCos module

Define if the connected encoder has an index or not.
For SinCos encoders the index is often called "reference".
Incremental encoders without index are often called AB encoders, while incremental encoders with index are often called ABI encoders.

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Hall sensor port	0x2112:23	USINT	8	No sensor port	0		readwrite	
				Sensor port 1	1			
				Sensor port 2	2			
				Sensor port 3	3			
				Sensor port 4	4			

				Sensor port 5 | 5				

Incremental encoder

This parameter is used to define to which Sensor port the Hall sensor is connected in case of using it together with an incremental encoder in order to have an approximate absolute position while the incremental encoder index is not yet found.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Sinewave cycles per revolution	0x2112:24	UINT	16	1		1		readwrite	

SinCos module

Number of sinewaves cycles in one motor revolution.
This value is going to be used to calculate the singleturn resolution (subindex 3 *Resolution*).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Sinewave res-	0x2112:25	USINT	8	8 | 13	Inc	readwrite	

olu-tion				16 12 32 11 64 10 128 9 256 8 512 7 1024 6 2048 5 4096 4 8192 3			

SinCos module

Select desired number of pulses in one sinewave.
This value is going to be used to calculate the singleturn resolution (sunindex 3 *Resolution*).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Sinusoidal output voltage	0x2112:26	USINT	8	up to 50 mVpp 15 up to 14		readwrite	

				60 mVpp			
				up to 75 mVpp	13		
				up to 0.1 Vpp	12		
				up to 0.12 Vpp	11		
				up to 0.14 Vpp	10		
				up to 0.15 Vpp	9		
				up to 0.2 Vpp	8		
				up to 0.28 Vpp	7		
				up to 0.4 Vpp	6		

				up to 0.5 Vpp: 5 up to 0.6 Vpp: 4 up to 0.75 Vpp: 3 up to 1 Vpp: 2 up to 1.2 Vpp: 1 up to 1.5 Vpp: 0			

SinCos module

Differential voltage in the Sin/Cos outputs peak to peak.
It is often abreviated as Vpp.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping

Filter	0x21-12:2-7	US-IN-T	8	Disabled: 0 Filter strength 2 (weakest): 2 Filter strength 3: 3 Filter strength 4: 4 Filter strength 5: 5 Filter strength 6: 6 Filter strength 7: 7 Filter strength 8: 8 Filter strength 9: 9		read-write	

				(stronge-st)				

REM 16MT

The filter with strength F of the sensor. The estimated step response time of the filter is Ts*2^F (Ts is the sampling time, ca. 70µs).

0x2113 Encoder 2 feedback

Feedback of the encoder configured as Encoder 2

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Raw position	0x2113:1	UDINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)
The position value directly read from the encoder.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Adjusted position	0x2113:2	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

The position value after applying the Singleturn offset and the Sensor polarity and the location of the index. This is an absolute value.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity	0x2113:3	DINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x2210 GPIO pin configuration

Configurable general purpose inputs and outputs with a 4 kHz update rate.

For hardware that supports it, each pin can be configured as follows:

Value	Definition
0	Disabled
2	[IN] General Purpose Input
3	[OUT] General Purpose Output
4	[IN] Home Switch Active High - Level Triggered
5	[IN] Home Switch Active Low - Level Triggered
6	[IN] Positive Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)
7	[IN] Positive Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)

8	[IN] Negative Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
9	[IN] Negative Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
10	[IN] Interlock (Disable Operation) Active Low - Level Triggered
11	[IN] Fault Reset - Edge Triggered
12	[IN] Interlock (Fault Reaction) Active Low - Level Triggered
13	[IN] Time Stamped Rising Edge - Edge Triggered
14	[OUT] Mimics Target Reached (tr) Bit of Statusword
15	[OUT] Mimics Internal Limit Active (ila) Bit of Statusword
16	[OUT] Timestamped Triggered Output
17	[OUT] Position Triggered Output (Pulse)
18	[OUT] Mimics Fault (f) Bit of Statusword
19	[OUT] Mimics Switched On (so) Bit of Statusword
20	[OUT] Mimics Ready to Switch On (rtso) Bit of Statusword
21	[OUT] Mimics Operation Enabled (oe) Bit of Statusword
22	[OUT] Mimics Voltage Enabled (ve) Bit of Statusword
23	[OUT] Mimics Switch On Disabled (sod) Bit of Statusword
24	[OUT] Mimics Quick Stop (qs) Bit of Statusword
25	[OUT] Mimics Warning (w) Bit of Statusword

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 1	0x2210:1	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Pos-ition Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

				Voltage Enabled [OUT] Switch On Disabled 23 [OUT] Quick Stop 24 [OUT] Warning 25 [OUT] NZR LED 26			
Pin 1 : DIO [With Pull Down Resistor]							
Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 2	0x2210:2	USINT	8	Disabled 0 [IN] General Purpose 2 [OUT] General Purpose 3 [IN] Home Switch (Act-		read-write	

				ive High)			
				[IN] Home Switch (Active Low)	5		
				[IN] Limit Switch Positive (Active High)	6		
				[IN] Limit Switch Positive (Active Low)	7		
				[IN] Limit Switch Negative (Active High)	8		
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault	11		

Reset (Active High)	
[IN] Interlock - Fault Reaction (Active Low)	12
[IN] Timestamped Rising Edge (Active High)	13
[OUT] Target Reached	14
[OUT] Internal limit Active	15
[OUT] Timestamped Triggered Output	16
[OUT] Position Triggered Output	17

(Pulse)	
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT] Voltage Enabled	22
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 2 : DIO [With Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 3	0x2210:3	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Position Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

				Voltage Enabled [OUT] Switch On Disabled: 23 [OUT] Quick Stop: 24 [OUT] Warning: 25 [OUT] NZR LED: 26			

Pin 3 : DIO [With Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 4	0x2210:4	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Act-		read-write	

				ive High)			
				[IN] Home Switch (Active Low)	5		
				[IN] Limit Switch Positive (Active High)	6		
				[IN] Limit Switch Positive (Active Low)	7		
				[IN] Limit Switch Negative (Active High)	8		
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault	11		

Reset (Active High)	
[IN] Interlock - Fault Reaction (Active Low)	12
[IN] Timestamped Rising Edge (Active High)	13
[OUT] Target Reached	14
[OUT] Internal limit Active	15
[OUT] Timestamped Triggered Output	16
[OUT] Position Triggered Output	17

(Pulse)	
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT] Voltage Enabled	22
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 4 : DIO [With Pull Down Resistor] (For Circulo : Only DO)

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 5	0x2210:5	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Position Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

				Voltage Enabled [OUT] Switch On Disabled: 23 [OUT] Quick Stop: 24 [OUT] Warning: 25 [OUT] NZR LED: 26			

Pin 5 : DIO [Without Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 6	0x2210:6	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Act-		read-write	

				ive High)			
				[IN] Home Switch (Active Low)	5		
				[IN] Limit Switch Positive (Active High)	6		
				[IN] Limit Switch Positive (Active Low)	7		
				[IN] Limit Switch Negative (Active High)	8		
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault	11		

				Reset (Active High)			
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target Reached	14		
				[OUT] Internal limit Active	15		
				[OUT] Timestamped Triggered Output	16		
				[OUT] Position Triggered Output	17		

(Pulse)	
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT] Voltage Enabled	22
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 6 : DIO [Without Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 7	0x2210:7	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Pos-ition Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

Voltage Enabled	
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 7 : DI [With Pull Down Resistor]

0x2211 GPIO output events

Configuration to set the value in the *Digital Output* object to the corresponding GPIO when a certain event happens.

The GPIO has to be enabled in *0x2210 GPIO*.

Configurable events are:

`Timestamp triggered`: Output at a certain timestamp.

`Target reached`: Output when target is reached. This mimics bit 10 (tr) of the Statusword.

`Internal limit active`: Output when internal limit is active. This mimics bit 11 (ila) of the Statusword.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Timestamp set-point	0x22-11:1	UDINT	32			0		read-write	Receive PDO (Outputs)
Timestamp at which the GPIO should trigger in case "Timestamp triggered" is selected. The unit is in 10 ns.									

0x2212 GPIO input actions

Configure an input GPIO to perform a certain action depending on its value.

The GPIO has to be enabled in *0x2210 GPIO*.

Configurable actions are:

`Timestamped rising edge`: Report timestamp of selected GPIO last rising edge.

`Disable operation`: Clear bit 3 (eo) of controlword when the selected GPIO is high.

`Quick stop`: Clear bit 2 (qs) of controlword when the selected GPIO is high.

`Fault reset`: Set bit 7 (fr) of controlword when the selected GPIO is high.

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Timestamp on rising edge	0x22-12:1	UDINT	32			0		read-only (default)	Transmit PDO (Inputs)
Timestamp at which the last rising edge happened in case "Timestamped rising edge" is selected. The unit is in 10 ns.									

0x2213 GPIO position trigger

Configure a GPIO to output a width-configurable pulse everytime certain encoder position is reached.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position setpoint	0x2213:1	DINT	32			0	Inc	read-write	Receive PDO (Outputs)
Position value at which the GPIO should be triggered.									

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Target encoder	0x2213:2	USINT	8	No encoder selected	0		read-write	
				Encoder 1	1			
				Encoder 2	2			
Encoder whose position is used for this feature.								

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

					a	a			
Pulse width	0x22-13:3	USINT	8			0	ms	read-write	
Width of the pulse when the GPIO is triggered.									

0x2214 GPIO global options

undefined

Name	Index:-Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Voltage level	0x2214:1	USINT	8	3.3 volts	1		read-write	
				5.0 volts	2			
Select the global voltage level of GPIO ports.								

0x2401 Analog input 1

Analog Input value of SOMANET Drive if available.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 1	0x2401:0	UINT	16			0		read-only (default)	Transmit PDO (Inpu-

									ts)

0x2402 Analog input 2

Analog Input value of SOMANET Drive if available.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 2	0x2402:0	UINT	16			0		read-only (default)	Transmit PDO (Inputs)

0x2403 Analog input 3

Analog Input value of SOMANET Drive if available.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 3	0x2403:0	UINT	16			0		read-only (default)	Transmit PDO (Inputs)

0x2404 Analog input 4

Analog Input value of SOMANET Drive if available.

Na-	Index:-	Ty-	Bit	Mi-	Ma-	Def-	Un-	Acce-	PDO

me	Sub	pe	Si-ze	n Da-ta	x Da-ta	ault Data	it	ss	Map-ping
Ana-log inpu-t 4	0x240-4:0	UI-NT	16			0		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x2611 Safety digital input diagnostics

Digital Inputs on Safety Module

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	Un-it	Acce-ss	PDO Map-ping
Inp-ut 1	0x261-1:1	BO-OL	1		1	0		read-only (defa-ult)	
Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	Un-it	Acce-ss	PDO Map-ping
Inp-ut 2	0x261-1:2	BO-OL	1		1	0		read-only (defa-ult)	

0x2701 Tuning command

This object is used in tuning mode to specify the parameter that is configured. Altering the default values is recommended for expert users only!

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Tuning command	0x2701:0	UDINT	32			0		read-write	Receive PDO (Outputs)

0x2702 Tuning status

For internal purposes only!

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Tuning status	0x2702:0	UDINT	32			0		read-write	Transmit PDO (Inputs)

0x2703 User MOSI

Reserved for SDK users.

Name	Index:Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Dat-a			ping
User MOSI	0x2703:0	UDINT	32			0		read-write	Receive PDO (Outputs)

0x2704 User MISO

Reserved for SDK users.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
User MISO	0x2704:0	UDINT	32			0		read-write	Transmit PDO (Inputs)

0x2705 Setup wizard completed

Used in OBLAC Drives for marking the successful completion of the setup guide. Intended for internal use only.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Setup wiz-	0x2705:0	USINT	8	0	3	0		read-write	

ard com-pleted									

0x603F Error code

Provides the error code of the last error which occurred. See object `0x203F Error report` for a more detailed description.

Error Code	Meaning
0x2220	Continuous over current (device internal)
0x2250	Short circuit (device internal)
0x2350	Load level fault (I2t, thermal state)
0x2351	Load level warning (I2t, thermal state)
0x3130	Phase failure
0x3131	Phase failure L1
0x3132	Phase failure L2
0x3133	Phase failure L3
0x3210	DC link over-voltage
0x3220	DC link under-voltage
0x3331	Field circuit interrupted
0x4210	Excess temperature device
0x4310	Excess temperature drive

Error Code	Meaning
0x5300	Operating unit
0x6010	Software reset (watchdog)
0x6320	Parameter error
0x7121	Motor blocked
0x7300	Sensor
0x7500	Communication

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Error code	0x603F:0	UINT	16			0		read-only (default)	Transmit PDO (Inputs)

0x6040 Controlword

This object shall indicate the received command controlling the PDS FSA. The bits 7, 3, 2, 1, and 0 shall be supported. The other bits may be supported.

15	11	10	9	8	7	6	4	3	2	1	0
ms		r	oms	h	fr	oms		eo	qs	ev	so

MSB	LSB

LEGEND: ms = manufacturer-specific; r = reserved; oms = operation mode specific; h = halt; fr = fault reset; eo = enable operation; qs = quick stop; ev = enable voltage; so = switch on

Bits of the *controlword*						
Command	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	Transitions
Shutdown	0	X	1	1	0	2, 6, 8
Switch on	0	0	1	1	1	3
Switch on + enable operation	0	1	1	1	1	3 + 4 (NOTE)
Disable voltage	0	X	X	0	X	7, 9, 10, 12
Quick stop	0	X	0	1	X	7, 10, 11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, 16
Fault reset	1	X	X	X	X	15
NOTE Automatic transition to Enable operation state after executing SWITCHED ON state functionality.						

Bits 9, 6, 5, and 4 of the controlword are operation mode specific. The halt function (bit 8) behaviour is operation mode specific. If the bit is 1, the commanded motion shall be interrupted, the PDS shall behave as defined in the halt option code. After releasing the halt function, the commanded motion shall be continued if possible.

The bit 10 is reserved for further use; it shall be set to 0. The bits 11, 12, 13, 14, and 15 are manufacturer-specific.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controlword	0x6040:0	UINT	16			0		read-write	Receive PDO (Outputs)

0x6041 Statusword

This object shall provide the status of the PDS FSA. The bits 10, 9, and 6 to 0 shall be supported. The other bits may be supported.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ms		oms		ila	tr	rm	ms	w	sod	qs	ve	f	oe	so	rtso
MSB															LSB

LEGEND: ms = manufacturer-specific; oms = operation mode specific; ila = internal limit active; tr = target reached; rm = remote; w = warning; sod = switch on disabled; qs = quick stop; ve = voltage enabled; f = fault; oe = operation enabled; so = switched on; rtso = ready to switch on

State coding

Statusword	PDS FSA state
xxxx xxxx x0xx 0000_b	Not ready to switch on
xxxx xxxx x1xx 0000_b	Switch on disabled
xxxx xxxx x01x 0001_b	Ready to switch on
xxxx xxxx x01x 0011_b	Switched on
xxxx xxxx x01x 0111_b	Operation enabled
xxxx xxxx x00x 0111_b	Quick stop active
xxxx xxxx x0xx 1111_b	Fault reaction active
xxxx xxxx x0xx 1000_b	Fault

If bit 4 (voltage enabled) of the statusword is 1, this indicates that high voltage is applied to the PDS.

If bit 5 (quick stop) of the statusword is 0, this indicates that the PDS is reacting on a quick stop request.

If bit 7 (warning) of the statusword is 1, this indicates the presence of a warning condition. Warning is not an error or fault (examples: temperature limit exceeded, job refused). The status of the PDS FSA is not changed. The cause of the warning is given in the Error report object (203Fh).

If bit 9 (remote) of the statusword is 1, this indicates that the controlword is processed. If it is 0 (local), this indicates that the controlword is not processed. Nevertheless, the PDS may provide actual values, and the PDS may accept COB for configuration data transmission for other parameter objects.

If bit 10 (target reached) of the statusword is 1, this indicates that the PDS has reached the set-point. The set-point is operation mode specific and is defined in detail in the corresponding clauses of this part of the IEC 61800-7 series. Bit 10 is also set to 1, if the operation mode has been changed. The change of a target value by software will alter this bit. If quick stop option code is 5, 6, 7 or 8, bit 10 is set to 1, when the quick stop operation is finished and the PDS is halted. If

halt occurred and the PDS has halted then bit 10 is set to 1, too.

Internal limit active (bit 11) indicates if there is an element that doesn't allow the motion to follow the target value. Several limits exist in the firmware. Max. torque (6072h) limits Torque demand (6074h), Max. motor speed (6080h) limits Velocity demand value (606Bh) and Profile velocity (6081h) in Profile position (PP) mode, Software position limit (607Dh) limits Position demand value (6062h) and position or velocity controller integral limit limits integral part of the PID controller.
Additional limit, Position range limit (607Bh), exists for Position demand value (6062h) in Profile position (PP) mode since demand value can't wrap around the range limit, when profiler is being used.

Operation mode specific bits of Statusword ("oms")

Bits	Cyclic synchronous position control	Cyclic synchronous velocity control	Cyclic synchronous torque control	Profile position mode	Profile velocity mode	Homing mode
bit 13 (0x2000)	Following error bit	not used		Following error bit	not used	Homing error bit
bit 12 (0x1000)	Target position ignored	Target velocity ignored	Target torque ignored	Set-point acknowledge bit	Speed bit	Homing attained bit

Bit 14 and bit 15 are manufacturer-specific.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Status-word	0x60-41:0	UI-NT	16			0		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x605A Quick stop option code

Indicates what action is performed when the quick stop function is executed. The slow down ramp is the deceleration value of the used mode of operations. The table below shows the value def-enitions for the function.

Value	Definition
0	Disable drive function
2	Slow down on quick stop ramp and transit into switch on disabled
6	Slow down on quick stop ramp and stay in quick stop active

Na-me	Index:-Sub	Ty-pe	Bit Si-ze	Options		U-nit	Acce-ss	PDO Map-ping
Qui-ck sto-p opti-on cod-e	0x605-A:0	UI-NT	16	Disable Drive Function	0		read-write	
				Slow down on quick stop ramp and	2			

				transit into switch on disabled			
				Slow down on quick stop ramp and stay in quick stop act-ive	6		

0x6060 Modes of operation

Use this object to request an operation mode.

NOTE This object shows only the value of the requested operation mode, the actual operation mode of the PDS is reflected in Object 0x6061:0.

Value	Definition
-2	Diagnostics mode
-1	Cogging compensation recording mode
+1	Profile position mode
+3	Profile velocity mode
+4	Torque profile mode
+6	Homing mode

Value	Definition
+8	Cyclic sync position mode
+9	Cyclic sync velocity mode
+10	Cyclic sync torque mode

The modes can be switched dynamically while in state `Operation enabled`.

Ensure that all required PDO data is available for the mode you are switching to.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Modes of operation	0x6060:0	SINT	8			0		read-write	Receive PDO (Outputs)

0x6061 Modes of operation display

Displays the actual operational mode.

Value	Definition
-2	Diagnostics mode
-1	Cogging compensation recording mode
0	No mode selected
+1	Profile position mode

Value	Definition
+3	Profile velocity mode
+4	Torque profile mode
+6	Homing mode
+8	Cyclic sync position mode
+9	Cyclic sync velocity mode
+10	Cyclic sync torque mode

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Modes of operation display	0x6061:0	SINT	8			0		read-only (default)	Transmit PDO (Inputs)

0x6062 Position demand value

Provides the value of the position sent to the position controller after position limits are checked. See `0x607D Software position limit`. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Position demand value	0x6062:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x6064 Position actual value

Provides the actual value of the position encoder. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position actual value	0x6064:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x6065 Following error window

The range of tolerated position values around the position demand value. If the position actual value is out of the following error window, a following error occurs. A following error may occur when a drive is blocked, an unreachable profile target is commanded, or when the controller is weakly tuned. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it). If the value of the following error window is 0xFFFFFFFF, the following error monitoring will be switched off.

Name	Index:Sub	Type	Bit Size	Min D-	Max D-	Default Data	Unit	Access	PDO Mapping

				at-a	at-a				
Following error window	0x6065:0	UDINT	32			4294967295	Inc	read-write	Receive PDO (Outputs)

0x6066 Following error time out

Configures the time to allow the controller to be outside the error window. After this timeout, bit 13 of the statusword will be set to 1. The units are in milliseconds.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Following error time out	0x6066:0	UINT	16			100	ms	read-write	Receive PDO (Outputs)

0x6067 Position window

Configures the size of the window by reflecting this value on either side of the target position. If the actual value of the position encoder is within this position window, the target position will be regarded as having been reached. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it). If the value of the position window is 0xFFFFFFFF, position window monitoring will be switched off.

Na-	Index-	Typ-	Bit	Mi-	M-	Default	U-	Acce-	PDO

me	:Sub	e	Si-ze	n D-at-a	ax D-at-a	Data	nit	ss	Map-ping
Pos-ition win-dow	0x60-67:0	UDI-NT	32			42949-67295	In-c	read-write	Rece-ive PDO (Out-puts)

0x6068 Position window time

Configures the time required for the position sensor to be within the position window before it's considered to be within the window. The units are in milliseconds.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Pos-ition win-dow time	0x606-8:0	UI-NT	16			0	m-s	read-write	Recei-ve PDO (Out-puts)

0x606B Velocity demand value

Provides the output value of the trajectory generator in velocity control, and the output of the position controller when using the cascaded controller. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-	Ma-x Da-	Def-ault Dat-	U-nit	Acce-ss	PDO Map-ping

				ta	ta	a			
Velocity demand value	0x606B:0	DINT	32			0	rpm	read-only (default)	Transmit PDO (Inputs)

0x606C Velocity actual value

Provides the actual velocity value derived either from the velocity sensor or the position sensor. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity actual value	0x606C:0	DINT	32			0	rpm	read-only (default)	Transmit PDO (Inputs)

0x606D Velocity window

Configures the size of the window by reflecting this value on either side of the target velocity. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it). If the value of the velocity window is 0xFFFF, velocity window monitoring will be switched off.

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Dat-a			ping
Velo-city win-dow	0x606-D:0	UI-NT	16			655-35	rp-m	read-write	Recei-ve PDO (Out-puts)

0x606E Velocity window time

Configures the time the velocity controller needs to be within the velocity window to consider it having reached the target. The units are in milliseconds.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Velo-city win-dow time	0x606-E:0	UI-NT	16			0	m-s	read-write	Recei-ve PDO (Out-puts)

0x606F Velocity threshold

Configures the velocity that is considered zero velocity. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Velo-	0x606-	UI-	16			0	rp-	read-	Recei-

city thres-hold	F:0	NT					m	write	ve PDO (Out-puts)

0x6070 Velocity threshold time

Configures the time required for the actual velocity to rise above the velocity threshold before the motor is considered to be moving. The units are in milliseconds.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Velo-city thres-hold time	0x607-0:0	UI-NT	16			0	m-s	read-write	Recei-ve PDO (Out-puts)

0x6071 Target Torque

Sets the desired torque for the torque controller in profile torque mode. The units are per thousand of rated torque.

Na-me	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Tar-get Tor-que	0x607-1:0	IN-T	16			0		read-write	Recei-ve PDO (Out-puts)

0x6072 Max torque

The configured maximum permissible torque in the motor. The value is given in per thousandth of rated torque.

$$\text{Max Torque} = \frac{\tau_{\text{max_Datasheet}}}{\tau_{\text{rated_Datasheet}}} \times 1000$$

For example, if the rated torque from the datasheet is 200 Nm and maximum torque is 600 Nm,

$$\text{Max Torque} = \frac{600\ \text{Nm}}{200\ \text{Nm}} \times 1000 = 3000$$

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max torque	0x6072:0	UINT	16	0	32767	3000		read-write	Receive PDO (Outputs)

0x6073 Max current

Indicates the configured maximum permissible torque-creating current in the motor. The units are per thousand of rated current.

$$\text{Max Current} = \frac{I_{\text{max_Datasheet}}}{I_{\text{rated_Datasheet}}} \times 1000$$

For example, if the `0x6075 Motor rated current` is 2 Amps and maximum current which can be applied for short duration is 6 Amps,

$$\text{Max Current} = \frac{6\ \text{Amps}}{2\ \text{Amps}} \times 1000 = 3000$$

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Dat-a	Dat-a			ping
Max cur-rent	0x607-3:0	UI-NT	16	0	500-00	300-0		read-write	Recei-ve PDO (Out-puts)

0x6074 Torque demand

Provides the output value of the torque trajectory generator. The units are per thousand of rated torque.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	U-nit	Acce-ss	PDO Map-ping
Torq-ue dem-and	0x607-4:0	IN-T	16			0		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x6075 Motor rated current

The motor rated current is taken from the motor datasheet. It needs to be configured in rms value. In the datasheet it could be defined in other units (peak value), therefore it needs to be converted into rms value before configuration. The units are in Milliampere [mA]. All relative current data refers to this value (such as `0x6073 Max current`).

Na-me	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Max Data	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping

Motor rated current	0x6075:0	DINT	32	1	1000000	1	mA	read-write	

0x6076 Motor rated torque

Indicates the nominal torque for the motor. It is taken from the motor's specifications. All relative torque data shall refer to this value (such as `0x6072 Max torque`). For linear motors, the object name is not changed, but the motor rated force value must be entered as multiples of mN (milli Newton). The value shall be given in mNm (milli Newton metre).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Motor rated torque	0x6076:0	DINT	32	1		1	mNm	read-write	

0x6077 Torque actual value

Provides the actual value of the torque, and is computed based on the currents measured in the motor. The units are per thousand of rated torque.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Torque actual value	0x6077:0	INT	16			0		read-only (default)	Transmit PDO (Inputs)

0x6079 DC link circuit voltage

Provides the bus voltage measured at the inverter. The units are in millivolts (mV).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
DC link circuit voltage	0x6079:0	UDINT	32			0	mV	read-only (default)	Receive PDO (Outputs)

0x607A Target position

Sets the desired position for all position modes. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Target pos-	0x607A:0	DINT	32			0	Inc	read-write	Receive PDO

ition									(Out-puts)

0x607B Position range limit

Reports the limits of the position sensor, and is updated when the encoder is reconfigured. On reaching or exceeding these limits, the sensor value will wrap automatically to the other end of the range. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Min position range limit	0x607B:1	DINT	32			2147483648	Inc	read-only (default)	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max position range limit	0x607B:2	DINT	32			2147483647	Inc	read-only (default)	Receive PDO (Outputs)

0x607C Home offset

Defines the difference between the zero position for the application and the machine home position (found during homing). During homing, the machine home position is found and once the homing is completed, the zero position is offset from the home position by adding the home offset to the home position. All subsequent absolute moves are taken relative to this new zero position. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Home offset	0x607C:0	DINT	32			0	Inc	read-write	Receive PDO (Outputs)

0x607D Software position limit

Defines the absolute position limits for the `0x607A Target position` and the `0x6064 Position actual value`. Every new target and actual position are checked against these limits. The limit positions shall be always relative to the machine home position. Limits are disabled by setting them to default values, min. limit to -2147483648 and max. limit to 2147483647. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Min position	0x607D:1	DINT	32			2147483648	Inc	read-write	Receive PDO

limit									(Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max position limit	0x607D:2	DINT	32			2147483647	Inc	read-write	Receive PDO (Outputs)

0x607E Polarity

Indicates if the velocity or position data are multiplied by 1 of by -1, effecitvely flipping the polarity of the command and feedback. The position polarity bit is used for profile position *(pp)* mode and cyclic sync position mode *(csp)*. The velocity polarity bit is used for profile velocity *(pv)* mode and cyclic sync velocity mode *(csv)*.

The valid values of this object are:

`Position polarity flipped` (128) - multiplies the position by -1
`Velocity polarity flipped` (64) - multiplies the velocity by -1
`Both are flipped` (192) - both velocity and position are multiplied by -1

The polarity flag has no influence on the homing mode.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Polarity	0x607E:0	USINT	8	None: 0 Position: 128		read-write	Receive PDO (Outputs)

				polarity flipped | Velocity polarity flipped | 64 Both flipped | 192				

0x6080 Max motor speed

Sets the max allowed speed for the motor in either direction. It is used to protect the motor and is taken from the motor data sheet. The value shall be given in RPM by default (see `0x60a9 SI unit velocity` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max motor speed	0x6080:0	UDINT	32	0		1000	rpm	read-write	Receive PDO (Outputs)

0x6081 Profile velocity

Configures the velocity target for the end of the acceleration ramp during profiled motion. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Dat-a			ping
Pro-file velo-city	0x608-1:0	UDI-NT	32			0	rp-m	read-write	Recei-ve PDO (Out-puts)

0x6083 Profile acceleration

Configures the acceleration used during profiled motion. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Profile accel-eration	0x60-83:0	UDI-NT	32			0	rp-m/-s	read-write	Rece-ive PDO (Out-puts)

0x6084 Profile deceleration

Configures the deceleration used during profiled motion. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-	M-ax D-	Def-ault Dat-	Un-it	Acce-ss	PDO Map-ping

				at-a	at-a	a			
Profile decel-eration	0x60-84:0	UDI-NT	32			0	rp-m/-s	read-write	Rece-ive PDO (Out-puts)

0x6085 Quick stop deceleration

Indicates the deceleration used to stop the motor when the quick stop function is activated. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Quick stop decel-eration	0x60-85:0	UDI-NT	32	1		100-00	rp-m/-s	read-write	Rece-ive PDO (Out-puts)

0x6086 Motion profile type

Indicates the type of motion profile used to perform profiled motion.

Value	Definition
0	Linear ramp (trapeziodal profile)

Na-	Index:-	Ty-	Bit	Mi-	Ma-	Def-	U-	Acce-	PDO

me	Sub	pe	Si-ze	n Da-ta	x Da-ta	ault Data	nit	ss	Map-ping
Moti-on pro-file type	0x608-6:0	IN-T	16	12-8	0	0		read-only (defa-ult)	Recei-ve PDO (Out-puts)

0x6087 Torque slope

Configures the rate of change of torque. The units are in per thousand of rated torque per second.

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Tor-que slop-e	0x608-7:0	UDI-NT	32			0		read-write	Recei-ve PDO (Out-puts)

0x6088 Torque profile type

Indicates the type of profile used to perform a torque profile.

Value	Definition
0	Linear ramp (trapezoidal profile)

Na-me	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-	Ma-x Da-	Def-ault Data	U-nit	Acce-ss	PDO Map-ping

				ta	ta				
Torque profile type	0x6088:0	INT	16			0		read-only (default)	Receive PDO (Outputs)

0x6091 Gear ratio

Configures the ratio of motor shaft revolutions to driving shaft revolutions. The sensor used for position control is defined to be on the driving shaft. All values are dimensionless.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Motor revolutions	0x6091:1	UDINT	32			1		read-write	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Shaft revolutions	0x6091:2	UDINT	32			1		read-write	Receive PDO (Outputs)

0x6098 Homing method

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Homing method	0x6098:0	SINT	8	[-4] Home on offset (0x2005:-4) before positive endstop	-4		read-write	Receive PDO (Outputs)
				[-3] Home on offset (0x2005:-4) before negative endstop	-3			
				[-2] Home on index pulse before positive endstop	-2			
				[-1] Home on index pulse before negative endstop	-1			

				No homing method assigned: 0 [1] Home on index pulse before positive limit switch rising edge: 1 [2] Home on index pulse before negative limit switch rising edge: 2 [3] Home on index pulse before positive home switch: 3			

				rising edge			
				[4] Home on index pulse after positive home switch rising edge	4		
				[5] Home on index pulse before negative home switch rising edge	5		
				[6] Home on index pulse after negative home switch rising edge	6		

				[7] Home on index pulse before home switch rising edge - Positive movement direction	7		
				[8] Home on index pulse after home switch rising edge - Positive movement direction	8		
				[9] Home on index pulse before home switch fall-	9		

				ing edge - Positive movement direction			
				[10] Home on index pulse after home switch falling edge - Positive movement direction	10		
				[11] Home on index pulse before home switch rising edge - Negative movement direction	11		

				[12] Home on index pulse after home switch rising edge - Negative movement direction	12		
				[13] Home on index pulse before home switch falling edge - Negative movement direction	13		
				[14] Home on index pulse after	14		

				home switch falling edge - Negative movement direction			
				[17] Home on negative limit switch rising edge (switch in inactive state)	17		
				[18] Home on positive limit switch rising edge (switch in inactive state)	18		
				[19] Home on	19		

				positive home switch rising edge (switch in inactive state)			
				[20] Home on positive home switch rising edge (switch in active State)	20		
				[21] Home on negative home switch rising edge (switch in inactive state)	21		
				[22]	22		

				Home on negative home switch rising edge (switch in active State)			
				[23] Home on home switch rising edge (switch in inactive state) - Positive move-ment dir-ection	23		
				[24] Home on home switch rising edge (switch in active	24		

				State)- Positive move- ment dir- ection			
				[25] Home on home switch fall- ing edge (switch in active State) - Positive move- ment dir- ection	25		
				[26] Home on home switch fall- ing edge (switch in inactive state) - Positive move- ment dir- ection	26		

				[27] Home on home switch rising edge (switch in inactive state) - Negative move-ment dir-ection	27		
				[28] Home on home switch rising edge (switch in active State) - Negative move-ment dir-ection	28		
				[29] Home on home switch fall-	29		

				ing edge (switch in active State) - Negative move-ment dir-ection			
				[30] Home on home switch fall-ing edge (switch in inactive state) - Negative move-ment dir-ection	30		
				[33] Home on next index pulse - Negative move-ment dir-ection	33		

				[34] Home on next index pulse - Positive movement direction: 34 [35] Home on current position (same as mode 37): 35 [37] Home on current position: 37				

0x6099 Homing speeds

Defines the speed at which the motor moves during homing. Two different values can be given; one is used while searching for a limit or home switch, the other is used while searching for zero position (i.e. the index pulse for QEI encoders). The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Speed during search for switch	0x6099:1	UDINT	32			0	rpm	read-write	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Speed during search for zero	0x6099:2	UDINT	32			0	rpm	read-write	Receive PDO (Outputs)

0x609A Homing acceleration

Configures the acceleration and deceleration to be used during a homing operation. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Homing acceleration	0x60-9A:0	UDINT	32			0	rpm/s	read-write	Receive PDO (Outputs)

0x60A9 SI unit velocity

Indicates the user-defined velocity units. The prefix is used by the firmware to scale the velocity; the second byte is fixed to revolutions (0xB4); the third byte can define minutes (0x47) or seconds (0x03).

The supported values are:

- `1 RPM` (value 0x00B44700 or 11814656) - default, but may not be precise enough for some applications.
- `0.1 RPM` (value 0xFFB44700 or 4290004736)
- `0.01 RPM` (value 0xFEB44700 or 4273227520)
- `0.001 RPM` (value 0xFDB44700 or 4256450304)
- `1 RPS` (value 0x00B40300 or 11797248)
- `0.1 RPS` (value 0xFFB40300 or 4289987328)
- `0.01 RPS` (value 0xFEB40300 or 4273210112)
- `0.001 RPS` (value 0xFDB40300 or 4256432896)

Configurations with a gearbox may find `0.001 RPM` or `0.001 RPS` is more appropriate.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
SI unit velocity	0x60-A9:0	UDINT	32	RPM 11814656 0.1 RPM 4290004-736 0.01 4273227-		read-write	

				<table><tr><td>RPM</td><td>520</td></tr><tr><td>0.00-1 RPM</td><td>4256450-304</td></tr><tr><td>RPS</td><td>11797248</td></tr><tr><td>0.1 RPS</td><td>4289987-328</td></tr><tr><td>0.01 RPS</td><td>4273210-112</td></tr><tr><td>0.00-1 RPS</td><td>4256432-896</td></tr></table>			

0x60B1 Velocity offset

Sets a feed-forward velocity as input to the velocity controller. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity offset	0x60B1:0	DINT	32			0	rpm	read-write	Receive PDO (Outputs)

0x60B2 Torque offset

Sets a feed-forward torque as input to the torque controller. The units are in per thousand of rated torque.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque offset	0x60B2:0	INT	16			0		read-write	Receive PDO (Outputs)

0x60F4 Following error actual value

Provides the actual value of the position following error. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Following error actual value	0x60F4:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x60FA Control effort

Provides a measure of the control effort (the output of the velocity control loop). The units are per thousand of rated torque.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Control effort	0x60FA:0	DINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x60FC Position demand internal value

Provides the output of the trajectory generator in profile position mode. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position demand internal value	0x60FC:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x60FD Digital inputs

Bit 0: Shows Negative Limit Switch Status (1 = Active, 0 = Inactive)

Bit 1: Shows Positive Limit Switch Status (1 = Active, 0 = Inactive)

Bit 2: Shows Home Switch Status (1 = Active, 0 = Inactive)

Bit 3: Shows Interlock Status (1 = Drive cannot be enabled, 0 = Drive can be enabled)

Bit 4 to 15: Reserved

The manufacturer-specific bits (16 to 31) show the state of the Drive's digital inputs.

Bit 16: Digital input 1

Bit 17: Digital input 2

:

Bit 31: Digital input 16

For example, if only Digital input 1 is high (and bits 0 to 3 are low), then object value will be 65536 (0x10000)

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Digital inputs	0x60FD:0	UDINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x60FE Digital outputs

The manufacturer-specific bits allow to set the state of the Drive's digital outputs.

Bit 0: Manual Brake Operation (0=Engage, 1=Disengage)

Bit 1 to 15: Reserved

Bit 16: Digital output 1

Bit 17: Digital output 2

:

Bit 31: Digital output 16

To change the state of an output, the corresponding bit in the **Bit mask** has to be set to 1.

For example, to drive Digtal Output 1 only:

First enable the corresponding mask bit by writting 65536 (0x10000) in **Bit mask**

Then

Writing 65536 (0x10000) in **Physical outputs** will drive the digital output 1 High.

Writing 0 (0x0) in **Physical outputs** will drive the digital output 1 Low.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Physical outputs	0x60FE:1	UDINT	32			0		read-write	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Bit mask	0x60FE:2	UDINT	32			0		read-write	Receive PDO (Outputs)

0x60FF Target velocity

Sets the target velocity and will be used as input to the trajectory generator. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Nam-	Index:-	Ty-	Bit	Mi-	Ma-	Def-	U-	Acce-	PDO

e	Sub	pe	Si-ze	n Da-ta	x Da-ta	ault Dat-a	nit	ss	Map-ping
Tar-get velo-city	0x60F-F:0	DI-NT	32			0	rp-m	read-write	Recei-ve PDO (Out-puts)

0x6502 Supported drive modes

This object indicates the supported drive modes. Each bit equal to 1 within this value indicates a mode which is supported.

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Sup-por-ted drive mode-s	0x65-02:0	UDI-NT	32			197-549		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x6621 Safety statusword

Status of STO and SBC. STO is the 8th bit of the first byte, SBC is the 8th bit of the second byte

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	U-nit	Acce-ss	PDO Map-ping

Byte 1	0x6621:1	USINT	8		1	0		read-only (default)	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Byte 2	0x6621:2	USINT	8		1	0		read-only (default)	

Communication Area

Index	Name
0x1000	Device type
0x1001	Error register
0x1005	COB-ID SYNC
0x1006	Communication cycle period
0x1008	Manufacturer device name
0x100A	Manufacturer software version
0x100C	Guard time
0x100D	Life time factor
0x1010	Store parameters

	0x1010:1	Save all parameters
0x1011		Restore default parameters
	0x1011:1	Restore all default parameters
	0x1011:2	Restore communication default parameters (disabled)
	0x1011:3	Restore application default parameters (disabled)
	0x1011:4	Restore manufacturer default parameters
0x1016		Consumer heartbeat time
	0x1016:1	Consumer heartbeat time
0x1017		Producer heartbeat time
0x1018		Identity object
	0x1018:1	Vendor-ID
	0x1018:2	Product Code
	0x1018:3	Revision number
	0x1018:4	Serial number
0x1019		Synchronous counter overflow value
0x1023		OS command
	0x1023:1	Command
	0x1023:2	Status
	0x1023:3	Response
0x1024		OS command mode

0x1400		Receive PDO1 parameter
	0x1400:1	COB-ID used by PDO
	0x1400:2	Transmission type
	0x1400:3	Inhibit time
0x1401		Receive PDO2 parameter
	0x1401:1	COB-ID used by PDO
	0x1401:2	Transmission type
	0x1401:3	Inhibit time
0x1402		Receive PDO3 parameter
	0x1402:1	COB-ID used by PDO
	0x1402:2	Transmission type
	0x1402:3	Inhibit time
0x1403		Receive PDO4 parameter
	0x1403:1	COB-ID used by PDO
	0x1403:2	Transmission type
	0x1403:3	Inhibit time
0x1600		Receive PDO1 mapping
	0x1600:1	Controlword
	0x1600:2	Modes of operation
	0x1600:3	Target Torque

	0x1600:4	Target position
	0x1600:5	Target velocity
	0x1600:6	Torque offset
	0x1600:7	Tuning command
0x1601		Receive PDO2 mapping
	0x1601:1	Physical outputs
	0x1601:2	Bit mask
0x1602		Receive PDO3 mapping
	0x1602:1	User MOSI
	0x1602:2	Velocity offset
0x1603		Receive PDO4 mapping
0x1800		Transmit PDO1 parameter
	0x1800:1	COB-ID used by PDO
	0x1800:2	Transmission type
	0x1800:3	Inhibit time
	0x1800:4	reserved
	0x1800:5	Event timer
0x1801		Transmit PDO2 parameter
	0x1801:1	COB-ID used by PDO
	0x1801:2	Transmission type

	0x1801:3	Inhibit time
	0x1801:4	reserved
	0x1801:5	Event timer
0x1802		Transmit PDO3 parameter
	0x1802:1	COB-ID used by PDO
	0x1802:2	Transmission type
	0x1802:3	Inhibit time
	0x1802:4	reserved
	0x1802:5	Event timer
0x1803		Transmit PDO4 parameter
	0x1803:1	COB-ID used by PDO
	0x1803:2	Transmission type
	0x1803:3	Inhibit time
	0x1803:4	reserved
	0x1803:5	Event timer
0x1A00		Transmit PDO1 mapping
	0x1A00:1	Statusword
	0x1A00:2	Modes of operation display
	0x1A00:3	Position actual value
	0x1A00:4	Velocity actual value

	0x1A00:5	Torque actual value
0x1A01		Transmit PDO2 mapping
	0x1A01:1	Analog input 1
	0x1A01:2	Analog input 2
	0x1A01:3	Analog input 3
	0x1A01:4	Analog input 4
	0x1A01:5	Tuning status
0x1A02		Transmit PDO3 mapping
	0x1A02:1	Digital inputs
0x1A03		Transmit PDO4 mapping
	0x1A03:1	User MISO
	0x1A03:2	Timestamp
	0x1A03:3	Position demand internal value
	0x1A03:4	Velocity demand value
	0x1A03:5	Torque demand
0x1C00		Sync Manager
	0x1C00:1	SyncMan 0
	0x1C00:2	SyncMan 1
	0x1C00:3	SyncMan 2
	0x1C00:4	SyncMan 3

0x1C10		SM 0 Assignment
0x1C11		SM 1 Assignment
0x1C12		SM 2 Assignment
	0x1C12:1	SubIndex 001
	0x1C12:2	SubIndex 002
	0x1C12:3	SubIndex 003
	0x1C12:4	SubIndex 004
0x1C13		SM 3 Assignment
	0x1C13:1	SubIndex 001
	0x1C13:2	SubIndex 002
	0x1C13:3	SubIndex 003
	0x1C13:4	SubIndex 004

Manufacturer Specific Area

Index		Name
0x2000		Command object (disabled)
0x2001		Commutation angle offset
0x2002		Position control strategy
0x2003		Motor specific settings
	0x2003:1	Pole pairs
	0x2003:2	Torque constant

	0x2003:3	Phase resistance
	0x2003:4	Phase inductance
	0x2003:5	Motor phases inverted
0x2004		Brake options
	0x2004:1	Pull voltage
	0x2004:2	Hold voltage
	0x2004:3	Pull time
	0x2004:4	Release strategy
	0x2004:5	Controller disable delay
	0x2004:6	DC bus voltage (deprecated)
	0x2004:7	Brake status
	0x2004:8	Minimum displacement (pin brake)
	0x2004:9	Maximum torque (pin brake)
	0x2004:10	Output voltage
	0x2004:11	Switching frequency
0x2006		Protection
	0x2006:1	Undervoltage setpoint
	0x2006:2	Overvoltage setpoint
	0x2006:3	Overcurrent setpoint
0x2008		Cogging torque compensation

	0x2008:1	State
	0x2008:2	Enabled
0x2009		Commutation offset
	0x2009:1	State
	0x2009:2	Applied percent of rated torque
	0x2009:3	Measurement method
	0x2009:4	Phasing controller Kp
	0x2009:5	Phasing controller Ki
	0x2009:6	Phasing controller Kd
0x200A		I2t
	0x200A:1	Enabled
	0x200A:2	Peak time
	0x200A:3	Percent max
	0x200A:4	Protection active
0x200B		Max power
0x2010		Torque controller
	0x2010:1	Controller Kp (disabled)
	0x2010:2	Controller Ki (disabled)
	0x2010:3	Controller Kd (disabled)
	0x2010:4	Field weakening enabled

	0x2010:5	Field weakening percentage
	0x2010:6	Field weakening starting speed
	0x2010:7	Field weakening ending speed
	0x2010:8	Commutation angle measurement delay
	0x2010:9	Switching frequency
	0x2010:10	Settling time
	0x2010:11	Damping ratio
	0x2010:12	Encoder source
0x2011		Velocity controller
	0x2011:1	Controller Kp
	0x2011:2	Controller Ki
	0x2011:3	Controller Kd
	0x2011:4	Controller integral limit
	0x2011:5	Encoder source
0x2012		Position controller
	0x2012:1	Position loop Kp
	0x2012:2	Position loop Ki
	0x2012:3	Position loop Kd
	0x2012:4	Position loop integral limit
	0x2012:5	Velocity loop Kp

	0x2012:6	Velocity loop Ki
	0x2012:7	Velocity loop Kd
	0x2012:8	Velocity loop integral limit
	0x2012:9	Encoder source
0x2013		Gain scheduling
	0x2013:1	Enabled
	0x2013:2	Threshold velocity 0
	0x2013:3	Position loop Kp 0
	0x2013:4	Position loop Ki 0
	0x2013:5	Position loop Kd 0
	0x2013:6	Position loop integral limit 0
	0x2013:7	Velocity loop Kp 0
	0x2013:8	Velocity loop Ki 0
	0x2013:9	Velocity loop Kd 0
	0x2013:10	Velocity loop integral limit 0
	0x2013:11	Threshold velocity 1
	0x2013:12	Position loop Kp 1
	0x2013:13	Position loop Ki 1
	0x2013:14	Position loop Kd 1
	0x2013:15	Position loop integral limit 1

	0x2013:16	Velocity loop Kp 1
	0x2013:17	Velocity loop Ki 1
	0x2013:18	Velocity loop Kd 1
	0x2013:19	Velocity loop integral limit 1
0x2014		Torque window
	0x2014:1	Value
	0x2014:2	Time
0x2015		Velocity feed forward
	0x2015:1	Gain
	0x2015:2	Cutoff frequency
0x2021		Velocity feedback filter
	0x2021:1	Type
	0x2021:2	Cutoff frequency
0x2022		Position feedback filter
	0x2022:1	Type
	0x2022:2	Cutoff frequency
0x2023		Notch filter
	0x2023:1	Enabled
	0x2023:2	Center frequency
	0x2023:3	Rejection band

0x2027		Control input FIR filter
	0x2027:1	Enabled
	0x2027:2	Order
0x2030		Core temperature
	0x2030:1	Measured temperature
0x2031		Drive temperature
	0x2031:1	Measured temperature
0x2038		External scaled measurement
	0x2038:1	Scaled measurement value
	0x2038:2	Analog input
	0x2038:3	Resistance
	0x2038:4	Constant a0
	0x2038:5	Constant a1
	0x2038:6	Constant a2
	0x2038:7	Constant a3
	0x2038:8	Constant a4
	0x2038:9	Constant a5
	0x2038:10	Cutoff frequency
	0x2038:11	Upper error threshold
	0x2038:12	Lower error threshold

0x203F		Error report
	0x203F:1	Description
0x2040		Input counter
0x20E1		High resolution data
	0x20E1:1	Data 1
	0x20E1:2	Data 2
	0x20E1:3	Data 3
	0x20E1:4	Data 4
0x20F0		Timestamp
0x20F2		Assigned name
0x20F3		DC synchronization
	0x20F3:1	DC pulse time
0x2110		Encoder 1 configuration
	0x2110:1	Sensor port
	0x2110:2	Type
	0x2110:3	Resolution
	0x2110:4	Zero velocity threshold
	0x2110:5	Polarity
	0x2110:6	Singleturn offset
	0x2110:7	Access signal type

	0x2110:8	Clock frequency
	0x2110:9	Frame size
	0x2110:10	Multiturn bits
	0x2110:11	Multiturn first bit position
	0x2110:12	Singleturn bits
	0x2110:13	Singleturn first bit position
	0x2110:14	Timeout
	0x2110:15	CRC polynomial
	0x2110:16	Maximum tbusy
	0x2110:17	Status bits active value
	0x2110:18	Parity type
	0x2110:19	First clock delay
	0x2110:20	Data ordering
	0x2110:21	Endianness
	0x2110:22	Index availability
	0x2110:23	Hall sensor port
	0x2110:24	Sinewave cycles per revolution
	0x2110:25	Sinewave resolution
	0x2110:26	Sinusoidal output voltage
	0x2110:27	Filter

0x2111		Encoder 1 feedback
	0x2111:1	Raw position
	0x2111:2	Adjusted position
	0x2111:3	Velocity
0x2112		Encoder 2 configuration
	0x2112:1	Sensor port
	0x2112:2	Type
	0x2112:3	Resolution
	0x2112:4	Zero velocity threshold
	0x2112:5	Polarity
	0x2112:6	Singleturn offset
	0x2112:7	Access signal type
	0x2112:8	Clock frequency
	0x2112:9	Frame size
	0x2112:10	Multiturn bits
	0x2112:11	Multiturn first bit position
	0x2112:12	Singleturn bits
	0x2112:13	Singleturn first bit position
	0x2112:14	Timeout
	0x2112:15	CRC polynomial

	0x2112:16	Maximum tbusy
	0x2112:17	Status bits active value
	0x2112:18	Parity type
	0x2112:19	First clock delay
	0x2112:20	Data ordering
	0x2112:21	Endianness
	0x2112:22	Index availability
	0x2112:23	Hall sensor port
	0x2112:24	Sinewave cycles per revolution
	0x2112:25	Sinewave resolution
	0x2112:26	Sinusoidal output voltage
	0x2112:27	Filter
0x2113		Encoder 2 feedback
	0x2113:1	Raw position
	0x2113:2	Adjusted position
	0x2113:3	Velocity
0x2210		GPIO pin configuration
	0x2210:1	Pin 1
	0x2210:2	Pin 2
	0x2210:3	Pin 3

	0x2210:4	Pin 4
	0x2210:5	Pin 5
	0x2210:6	Pin 6
	0x2210:7	Pin 7
0x2211		GPIO output events
	0x2211:1	Timestamp setpoint
0x2212		GPIO input actions
	0x2212:1	Timestamp on rising edge
0x2213		GPIO position trigger
	0x2213:1	Position setpoint
	0x2213:2	Target encoder
	0x2213:3	Pulse width
0x2214		GPIO global options
	0x2214:1	Voltage level
0x2401		Analog input 1
0x2402		Analog input 2
0x2403		Analog input 3
0x2404		Analog input 4
0x2611		Safety digital input diagnostics
	0x2611:1	Input 1

	0x2611:2	Input 2
0x2701		Tuning command
0x2702		Tuning status
0x2703		User MOSI
0x2704		User MISO
0x2705		Setup wizard completed

Profile Specific Area

Index	Name
0x603F	Error code
0x6040	Controlword
0x6041	Statusword
0x605A	Quick stop option code
0x6060	Modes of operation
0x6061	Modes of operation display
0x6062	Position demand value
0x6064	Position actual value
0x6065	Following error window
0x6066	Following error time out
0x6067	Position window
0x6068	Position window time

0x606B		Velocity demand value
0x606C		Velocity actual value
0x606D		Velocity window
0x606E		Velocity window time
0x606F		Velocity threshold
0x6070		Velocity threshold time
0x6071		Target Torque
0x6072		Max torque
0x6073		Max current
0x6074		Torque demand
0x6075		Motor rated current
0x6076		Motor rated torque
0x6077		Torque actual value
0x6079		DC link circuit voltage
0x607A		Target position
0x607B		Position range limit
	0x607B:1	Min position range limit
	0x607B:2	Max position range limit
0x607C		Home offset
0x607D		Software position limit

	0x607D:1	Min position limit
	0x607D:2	Max position limit
0x607E		Polarity
0x6080		Max motor speed
0x6081		Profile velocity
0x6083		Profile acceleration
0x6084		Profile deceleration
0x6085		Quick stop deceleration
0x6086		Motion profile type
0x6087		Torque slope
0x6088		Torque profile type
0x6091		Gear ratio
	0x6091:1	Motor revolutions
	0x6091:2	Shaft revolutions
0x6098		Homing method
0x6099		Homing speeds
	0x6099:1	Speed during search for switch
	0x6099:2	Speed during search for zero
0x609A		Homing acceleration
0x60A9		SI unit velocity

0x60B1		Velocity offset
0x60B2		Torque offset
0x60F4		Following error actual value
0x60FA		Control effort
0x60FC		Position demand internal value
0x60FD		Digital inputs
0x60FE		Digital outputs
	0x60FE:1	Physical outputs
	0x60FE:2	Bit mask
0x60FF		Target velocity
0x6502		Supported drive modes
0x6621		Safety statusword
	0x6621:1	Byte 1
	0x6621:2	Byte 2

Process Data Objects (PDO)

Process Data Objects (PDO) are used to cyclically transfer information between master and slaves.

There is a standard set of PDO which consists of input objects (TxPDO) and output objects (RxPDO):

- Standard RxPDO mapped Objects
- Standard TxPDO mapped Objects

You can also create your own set of PDO to include the objects which are more suitable for your needs.

- Configuring custom PDO mappings

Standard RxPDO mapped Objects

Index	Name	Bit Length	Bit Offset
0x6040	Controlword	16	RxPDO Mapping 1
0x6060	Modes of operation	8	RxPDO Mapping 1
0x6071	Target Torque	16	RxPDO Mapping 1
0x607A	Target position	32	RxPDO Mapping 1
0x60FF	Target velocity	32	RxPDO Mapping 1
0x60B2	Torque offset	16	RxPDO Mapping 1
0x2701	Tuning command	32	RxPDO Mapping 1
0x60FE:1	Physical outputs	32	RxPDO Mapping 1
0x60FE:2	Bit mask	32	RxPDO Mapping 1
0x2703	User MOSI	32	RxPDO Mapping 1
0x60B1	Velocity offset	32	RxPDO Mapping 1

Standard TxPDO mapped Objects

Index	Name	Bit Length	Bit Offset
0x6041	Statusword	16	TxPDO Mapping 1
0x6061	Modes of operation display	8	TxPDO Mapping 1

0x6064	Position actual value	32	TxPDO Mapping 1
160x606C	Velocity actual value	32	TxPDO Mapping 1
0x6077	Torque actual value	16	TxPDO Mapping 1
0x2401	Analog input 1	16	TxPDO Mapping 1
0x2402	Analog input 2	16	TxPDO Mapping 1
0x2403	Analog input 3	16	TxPDO Mapping 1
0x2404	Analog input 4	16	TxPDO Mapping 1
0x2702	Tuning status	32	TxPDO Mapping 1
0x60FD	Digital inputs	32	TxPDO Mapping 1
0x2704	User MISO	32	TxPDO Mapping 1
0x20F0	Timestamp	32	TxPDO Mapping 1
0x60FC	Position demand internal value	32	TxPDO Mapping 1
0x606B	Velocity demand value	32	TxPDO Mapping 1
0x6074	Torque demand	16	TxPDO Mapping 1

Configuring custom PDO mappings

EtherCAT PDOs are transmitted cyclically in a fixed length. The standard set of PDO can be sent at up to 4000 Hz.

Each PDO mapping can be used for different use cases, for example a diagnostics mapping can be configured in one PDO mapping and this can be discarded for operation.

Mappings can also be added for different drives.

A total of four PDO mappings can be defined with up to 10 objects in each direction.

The defined mappings are configured in objects 0x1A00 - 0x1A03 (TxPDO Mapping) and 0x1600 - 0x1603 (RxPDO Mapping).

The PDO mappings that should be used are defined in the objects 0x1C12 (Incoming: SM Assignment 2) and 0x1C13 (Outgoing: SM Assignment 3).

> Note:
> Since the SM Assignment Mailboxes work as a triple buffer, only 0.3 kByte of data can be used in each PDO mapping.

Changing the content of an object

> Note:
> The content of a mapping object can only be changed in Pre-Operational state.

To change the content of an object:

1. Disable the object by setting its subindex 0 to 0
2. Enter the objects that should be mapped to the mapping entry subindices 1 to n
3. Set subindex 0 to the number n of used entries

> Note:
> If the subindex 0 is not 0 a write access to any other subindex will only be possible if the entered value equals the stored value. This can be used to check the device configuration against the master configuration.

More details can be found in our System Integration Guide

Parameters

Object ID	Name	Description
0x1600	RxPDO Mapping 1	Used for defining the incoming PDO mapping 1
0x1601	RxPDO Mapping 2	Used for defining the incoming PDO mapping 2
0x1602	RxPDO Mapping 3	Used for defining the incoming PDO mapping 3
0x1603	RxPDO Mapping 4	Used for defining the incoming PDO mapping 4
0x1A00	TxPDO Mapping 1	Used for defining the outgoing PDO mapping 1
0x1A01	TxPDO Mapping 2	Used for defining the outgoing PDO mapping 2
0x1A02	TxPDO Mapping 3	Used for defining the outgoing PDO mapping 3
0x1A03	TxPDO Mapping 4	Used for defining the outgoing PDO mapping 4
0x1C00	Sync Manager	This object handles the SM assignment
0x1C10	SM 0 Assignment	This object defines the master to slave communication (SDO and FoE)
0x1C11	SM 1 Assignment	This object defines the slave to master communication (SDO and FoE)
0x1C12	SM 2 Assignment	This object object defines the incoming PDO mappings to be used

Object ID	Name	Description
0x1C13	SM 3 Assignment	This object object defines the outgoing PDO mappings to be used

Device Information

- Saving and restoring configurations:
  Configurations can be saved for later use and default configurations can be restored.

These Objects specify how to communicate with a device and how to configure it:

Object ID	Name	Description
0x100A	Manufacturer Software Version	This object shows the Software Version
0x1000	Device Type	This object describes the type of the logical device and its functionality.
0x1008	Manufacturer Device Name	Indicates the name given by Synapticon.
0x1018	Identity	Identification of the EtherCAT device.

Saving and restoring configurations

Overview

This function allows saving and restoring the default configuration or restoring the factory configuration of the drive. The configuration files require a password to avoid accidental changes.

> Note:
> Saving factory defaults is not possible via firmware command. This must be performed offline. In case needed please contact support@synapticon.com for details.

The default values are loaded at startup, first the factory defaults (.factory_config), then the user configuration (config.csv).

- .factory_config contains the original configuration when the drive was shipped.
- config.csv is updated every time the “save to device” button is pressed or the parameters are saved by object 0x1010:1. This stores all parameters at once.

Use case:

Restoring configurations can be used for example for testing parameters without saving and safely going back to the last stable value. Saving the config.csv file can be used:

- as a backup
- for comparing different configurations
- for transferring the configuration to another drive.

The format for factory and user configuration files is CSV, so that it can be modified easily by text editor or spreadsheet. Lines of the config.csv file can simply be copied over into the factory defaults file without modification. The .factory_config file is protected and can not be changed or deleted by regular file operations

Attention:

The config.csv contains motor-individual parameters like the commutation offset or cogging compensation. Moving the config.csv from one drive to the other may require adjusting some parameters individually.

Attention:

The values loaded into the object dictionary are not checked for validity. Ensure that the values are correct when the parameters are set manually, otherwise malfunctions or erratic behavior might be caused that could lead to serious accidents.

When the drive first starts, it locates the .factory_config file and loads it into the object dictionary. If it does not exist, this step is skipped. Secondly, it locates the config.csv and loads the parameters into the object dictionary.

If the config.csv file doesn't exist, the drive will be configured as defined by the .factory_config.

Usage

> Danger:
>
> The drive must be disabled, so that the operation can be carried out safely.

Restore configuration

To restore parameter values from either file into the object dictionary, object 0x1011 (Restore default parameters) has to be triggered. Two subindexes are relevant:

1. Restore all (user) default parameters at subindex 1
2. Restore manufacturer (factory) default parameters at subindex 4

To avoid accidentally restoring the default parameters, restoring is only executed when a password is written to subitem 0x1011:1: 0x64616f6c (daol in Hexcode)

MSB			LSB
d	a	o	l
64h	61h	6Fh	6Ch

> Note:
>
> All parameters that are not stored will be discarded on power off since all parameters are restored automatically at startup.

> Note:
>
> It takes several milliseconds to restore the configuration file.

Save configuration

User defaults - config.csv

To save the state of the object dictionary for loading later, use object Store parameters 0x1010 subitem 1 (Save all parameters).

To protect the file against accidental tampering, a password must be written to the subitem 0x1010:1: 0x65766173 (*evas* in Hexcode)

MSB			LSB
e	v	a	s
65h	76h	61h	73h

After this operation all configuration state parameters are saved to the config.csv file located in the flash memory. It can be recovered via the filesystem using any communication protocol that supports file operations. For details, please refer to Drive operations through File Access over EtherCAT (FoE) .

It's also possible to save the configuration file in OBLAC Drives when clicking on the button "Save parameters".

> Note:
>
> It takes about 3 seconds to save the configuration file.

Limitations

- The drive must be disabled for saving or restoring parameters.
- This feature is supported in all modes of operation.

Parameters

Object ID	Name	Description
0x1010	Store parameters	Triggers storing user-defined parameters to the flash memory of the drive.
0x1011	Restore default parameters	Loads either factory settings or user defined parameters from the drive.

Software Reference 5.1

Migration guide from v4.x to 5.1

Firmware Overview Functional description

CANopen over EtherCAT Description of the working principle

Actuator configuration Contains information about motor and gear settings, as well as a description of the brake release feature

Motor and gear settings Description of the motor and gear parameters

Generic brake output voltage Phase D can be used for a single phase device or custom brake options

Brake configuration Description of the brake release feature

User defined velocity units Users can change the default velocity unit for higher precision

Commutation Offset Detection Aligns the motor's electrical angle with the position sensor feedback for motion control

Power limit Limits the mechanical output power of the motor

Encoders and I/O Description of analog and digital inputs and outputs and encoder configuration

Analog In How to set up analog encoders directly or using external scaled measurement

Analog Input Scaling Allows a nonlinear conversion of an analog voltage using a 5th order polynomial function

Digital I/O Configuration of digital I/Os as input or output

Digital GPIO Assigns actions to changes in the Digital Inputs and triggers Digital Outputs

Encoder configuration Description of the most important encoder parameters

Encoder sampling frequency The sampling frequency of certain encoders can be selected

Commutation with Hall and ABI combined Hall sensors and ABI encoders can be combined for commutation

Input counter Allows counting pulses in a particular pin of an encoder connector

Addressable LED control Allows controlling a strip of addressable LEDs through the Digital I/O port

Protection Users can define limits for undervoltage, overvoltage and overcurrent

Errors and warnings A list of all standard and SOMANET specific errors

Error codes thrown by the firmware SOMANET throws firmware errors as specified in CiA 402

Error report object Each occurring error is described in the Error Report Object

Remedies for protection errors How to resolve issues with over-current, over-voltage, under-voltage and over-temperature

Logging Logs are provided at every state change as well as at each occurring error or warning

Motion control Details about motion functions available via EtherCAT

Control Loops Overview of the control loops used in SOMANET

Current and Torque Control Loop Details and working principle about the current control loop

Velocity Control Loop Details about the velocity loop structure and the applied filters

Position Control Loop Details about the cascaded position loop structure and the applied filters

Dual Loop Cascaded Position Control For precise position control, it is possible to use two encoders for each cascade

Modes of operation Overview of the modes available in SOMANET

Cyclic modes In cyclic modes, the master generates and sends cyclic data frames to the drive

Cyclic synchronous torque control (CST) In torque control mode, the master sends desired torques to the drive

Cyclic synchronous velocity control (CSV) In velocity control modes, the master sends desired velocities to the drive

Cyclic synchronous position control (CSP) In position control modes, the master sends desired positions to the drive

Profile modes In profile modes, the trajectory is generated on the drive

Profile position mode In profile position mode, the trajectory to a target point is generated on the drive

Profile velocity mode In profile velocity mode, the motor will rotate at a target velocity

Profile torque mode In profile torque mode, the drive sends a target torque to the motor

Homing modes SOMANET offers various methodsto restore the position to a previously defined location

Diagnostics opmode Used for triggering diagnostic OS commands

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

Switching the modes Describes how the modes can be selected

Control Loop Tuning Each motor needs to be tuned before it is operational. Synapticon provides an easy-to-use Auto-Tuning feature but all drives can also be tuned manually

Extended control functionalities These features can be used to enhance performance

Cogging compensation Removes ripples caused by magnetic interaction between motor rotor and stator

Field weakening The velocity range of electric motors can be extended by weakening the magnetic field of the rotor linearly over the speed range

Anti-windup control An Anti-Windup algorithm is used to prevent the Integrator part to accumulate beyond acceptable boundaries

Gain scheduling Applies controller gains, which will change dynamically regarding the current speed

Motor overload protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating

Velocity feed forward To increase the velocity of the robot's reaction, the target position signal can bypass the position controller

Torque offset A desired offset torque can be added to the reference torque

Velocity offset An additional velocity offset can be used to improve the positioning performance

Filtering Digital filters can be used to remove unwanted frequencies such as noise or nonlinear effects

Position and Velocity Feedback Low-pass filters Reduce encoder noise and effects of encoder nonlinearity

Torque loop input shaping filter Suppress excitation of resonance frequencies

Command smoothing and Interpolation Successive setpoints can be smoothed, so that low command resolution and jitter have a reduced effect on the control

Control supervision Allows to supervise the work of the control loops

Feed constant For linear motion, the values for position, velocity and acceleration can be shown in linear units

Store home position The home position value can be saved in the Object Dictionary

Distributed clocks (DC clocks) Allows to synchronize the system time for enhanced performance

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

OS Command Offers commands that can't be executed by single SDO operations

Quick stop Stops the motor quickly in a controlled way

Object dictionary The object dictionary is defined in IEC 61800-7-201

All objects A list of the entire Object Dictionary

Communication area Objects needed for communication

Manufacturer specific area Complementary objects for enhanced performance and additional features

Profile specific area Objects that are used by the CANopen device profile for drives and motion control (CiA 402)

Process Data Objects (PDO) These objects are transferred cyclically to each slave

Standard RxPDO mapped objects List of outputs that are mapped in the standard configuration of SOMANET CiA 402

Standard TxPDO mapped objects List of inputs that are mapped in the standard configuration of SOMANET CiA 402

Configuring custom PDO mappings Choose your own objects for the PDO mapping

Device info These objects describe your Drive system for internal identification

Saving and restoring configurations Configurations can be saved for later use and default configurations can be restored

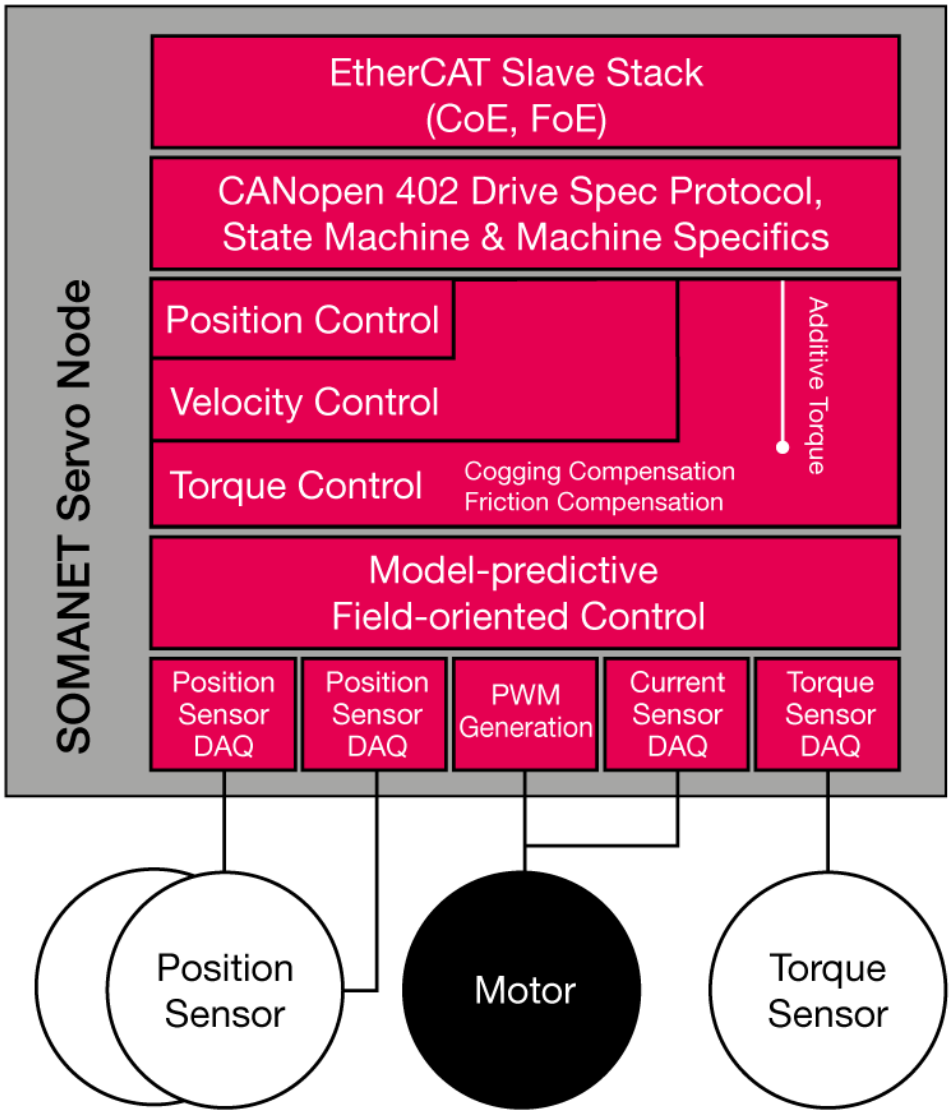
Changelog v5.0

Firmware Overview

All SOMANET servo drives support both cyclic synchronous modes of operation **CSP, CSV, CST** optimized for use in robotic applications. and profile modes **PP, PV, PT** for point-to-point applications. Firmware updates and storing of configuration files is possible via File over EtherCAT **(FoE)**. Multiple concurrent encoders (for example **BiSS-C, SSI, Incremental, Digital Halls**) are supported.

An object dictionary according to **CiA 402** is available, complemented with manufacturer specific objects to expand the feature set.

The following diagram depicts the basic architecture of the servo drive system:



Functionality description

Cyclic Synchronous Modes:

- CSP
- CSV
- CST

Profile Modes:

- Profile position mode
- Profile velocity mode
- Profile torque mode

Homing Modes that support limit and home switches

Set of control architectures:

- Field-oriented current control
- PI and model-predictive current control
- P, PI, PID and cascaded position control
- P, PI, PID velocity control
- Dual loop cascaded position control
- Controller Auto-Tuning and manual tuning

Various configurable signal processing options (low pass filters, top notch filters)

Encoders configuration for all ports: BiSS-C, Hall, ABI (QEI) / ABZ Encoder, SSI , A-Format

Access to Analog Inputs and Digital IO

Digital I/Os can be configured to timestamp inputs and outputs on rising edge

Optimized power output by 3rd harmonic injection

Synchronization with Distributed Clocks

Simple commissioning and tuning with OBLAC Drives

Extended control functionalities:

- Cogging torque compensation
- Field weakening
- Anti-Windup Control
- Gain Scheduling
- Motor Overload Protection (i2t)
- Feed Forward Control
- Torque Offset
- Velocity Offset
- Control supervision
- Command smoothing and Interpolation

CANopen over EtherCAT (CoE)

CANopen is a fieldbus system and device profile specification for embedded systems used in automation. Among others CANopen defines communication models like master/slave, where the master sends or requests data from the slaves. CANopen over EtherCAT (CoE) incorporates specialized profiles for motion control and supports a standardized set of commands to be used for motion control. This set of commands is known as CiA 402 and is defined in the IEC standard IEC 61800-7-201:2015 for adjustable speed electrical power drive systems.

If you are using TwinCAT as your controller, you will need to import the SOMANET CiA 402 ESI (EtherCAT Slave Information) XML file to your project.

Drive State Machine (CiA 402)

Each CiA 402 device is controlled by the State Machine.

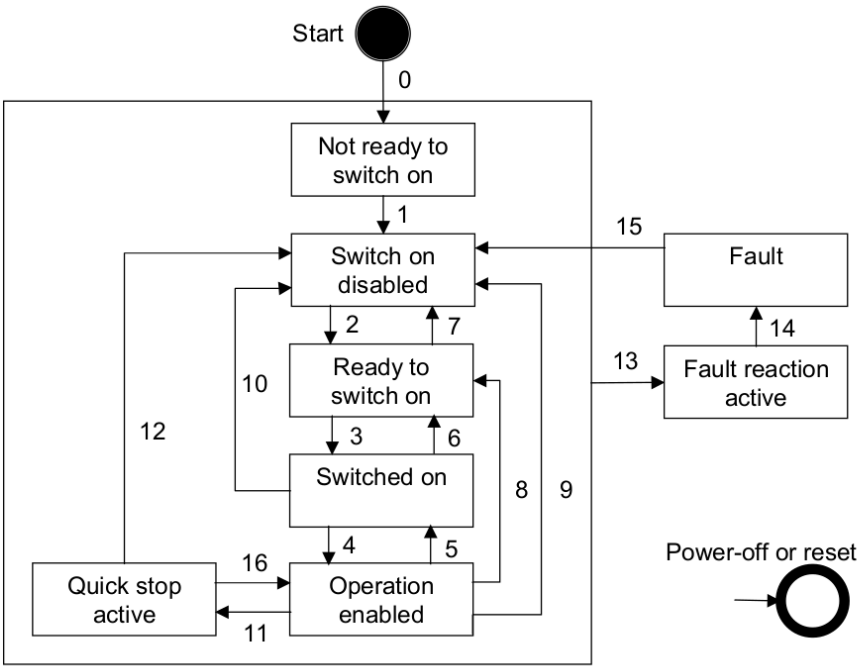


Image 1 - The transitions of the system's states according to IEC 61800-7-201

The function that are available in each state are given in the table below:

Function	FSA States							
	Not ready to switch on	Switch on disabled	Ready to switch on	Switched on	Operation enabled	Quick Stop active	Fault reaction active	Fault
Brake applied, if present	Yes	Yes	Yes	Yes	No	No	No	Yes
Drive function enabled	No	No	No	No	Yes	Yes	Yes	No
Configuration allowed	Yes	Yes	Yes	Yes	No	No	No	Yes

The devices of the motion control system are configured by the Service Data Objects (SDO) while the real-time process data is transferred by Process Data Objects (PDO).

PDO

PDO (Process data objects) are used in EtherCAT to cyclically transfer in real-time control (e.g. target velocity) and status (e.g. current velocity) variables. These variables are identified by their Index and Subindex in the object dictionary. With the help of the PDO mapping, the objects from the object dictionary are assigned to either receive PDO (RxPDO, Master to Slave direction) or transmit PDO (TxPDO, Slave to Master direction). The standard mapping can be changed to a custom mapping to accommodate specific needs.

PDO data is internally updated by the slave (servo drive) at a rate of up to 4 kHz. The master can access the drive at 4 kHz maximum.

For more information about PDO refer to ETG1000.5 and ETG1000.6

PDO Configuration

SOMANET supports EtherCAT PDO Configuration for up to 20 objects on 4 maps. This allows to use up to 80 objects per direction.
PDO Config allows to change the mapping of the PDO mappings itself. Every PDO mappable object from the object dictionary can be added into a PDO mapping. It is possible to add up to 20 objects to one PDO mapping

SDO

SDO (Service Data Objects) are used to exchange parameters between master and slave devices in a EtherCAT Network. This isn't done in real-time. They provide read and write access to all entries in the device object dictionary.

Typical examples are:

- Configuring the Drive system
- Reading error codes

For more information about SDO refer to ETG1000.5 and ETG1000.6.

Actuator configuration

Motor and gear settings

Generic brake output voltage

Brake configuration

User defined velocity units

Commutation Offset Detection

Power limit

Motor and gear settings

Motor Configuration

The servo drive needs several motor parameters to operate correctly. They can usually be found in the motor data sheet and are entered in the corresponding objects via the "OBLAC Drives" configuration wizard.

Pole pairs

In a three-phase PM motor, the number of poles depends on the rotor design and shows how many "south" and "north" magnetic poles exist. These magnetic poles are created by permanent magnets which are built into the rotor structure.

This number can be found in the motor's data sheet.

> Attention:
>
> If the number given in the data sheet is not designated as Pole pairs but Poles, it must be converted:
>
> $$\text{Number of pole pairs} = \frac{1}{2} \cdot \text{Number of poles}$$

Torque constant

The torque constant is the ratio between generated torque and the phase current amplitudes. The torque constant unit is in [µNm / A_{rms}]. The motor torque constant can be found in the motor's data sheet.

If in the motor data sheet, the unit of torque constant is in [Nm / A_{peak}], the following equation can be used to convert the unit of torque constant to [µNm / A_{rms}]:

$$K_t[\frac{Nm}{A_{rms}}] = \sqrt{2} \cdot K_t[\frac{Nm}{A_{peak}}]$$

Phase resistance

For a three-phase motor with star winding configuration, the phase resistance represents the resistance between motor terminals and the motor neutral point. If only phase to phase resistance is available in the motor data sheet, the value for phase resistance is half of the value phase to phase resistance in the motor data sheet:

> Attention:
>
> $$\text{Phase resistance} = \frac{1}{2} \cdot \text{Phase to phase resistance}$$

Phase resistance of delta winding motors

The software assumes that the motor windings are in a star configuration. For a "delta" winding configuration, the values which should be entered in the configuration must be converted to the equivalent star winding configuration.

If the value for phase resistance is available in the motor data sheet, the motor phase resistance for the equivalent star configuration equals to:

$$\text{Phase resistance}_{\text{equivalent star configuration}} = \frac{2}{3} \cdot \text{Phase resistance}_{\text{motor data sheet}}$$

Phase to phase resistance of delta winding motors

If the value for phase to phase resistance (also designated as terminal to terminal resistance or line to line resistance) is available in the motor data sheet, the motor phase resistance for the equivalent star configuration equals to:

$$\text{Phase resistance}_{\text{equivalent star configuration}} = \frac{1}{2} \cdot \text{Phase to phase resistance}_{\text{motor data sheet}}$$

Phase inductance

For a three-phase motor with a star winding configuration, the phase inductance represents the inductance between the motor terminals and the motor's neutral point. If only phase to phase inductance is available in the motor data sheet, the value for phase inductance is half of the phase to phase inductance value in the motor data sheet.

> Attention:
> $$\text{Phase inductance} = \frac{1}{2} \cdot \text{Phase to phase inductance}$$

Phase to phase inductance of delta winding motors

If the value for phase to phase inductance (also designated as terminal to terminal inductance or line to line inductance) is available in the motor data sheet, the motor phase inductance for the equivalent star configuration equals to:

$$\text{Phase inductance}_{\text{equivalent star configuration}} = \frac{1}{2} \cdot \text{Phase to phase inductance}_{\text{motor data sheet}}$$

> Note:
> Sometimes motor data sheets do not provide the phase inductance values but instead they provide the inductance values in direct and quadrature directions (Ld and Lq). In this case, the value of phase inductance can be calculated from the following equation:
> $$\text{Phase inductance} = \frac{1}{3} \cdot (\text{Ld} + \text{Lq})$$

Errors in phase resistance and inductance

Wrong values of the parameters phase resistance or phase inductance will affect the quality of the current loop controllers:

- narrower bandwidth
- less stability margins

> Note:
>
> Internal algorithms prevent degradation of the steady state operation when the error is less than 20% of the exact value.

Motor phases inverted

This parameter depends on the order in which the motor terminals are connected to the inverter terminals. It can be either “normal” or “inverted”. Details can be found in Commutation Offset Detection. This parameter is detected automatically when the sensor offset measurement procedure is triggered.

If Motor phases inverted is set wrong (manually), the following problems might appear:

- Wrong motion polarity (positive “torque target” resulting in negative “motor generated torque”).
- Low efficiency of motor

PWM frequency

The PWM frequency can be changed in object 0x2010:9 Torque controller: Switching frequency.

- Enter the value 0 for 16 kHz
- Enter the value 1 for 32 kHz
- Enter the value 2 for 64 kHz

By default, the switching frequency is set to 16 kHz. Higher switching frequency values will result in higher switching losses in the inverter and less copper and iron losses in the motor. Moreover, higher switching frequencies will result in less ripples in currents and voltages of the DC link which results in less ripple amplitudes in the currents and voltages of the power supply.

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

> Note:
> If the motor time constant is bigger than 100 ms, it's recommended to use 16 kHz. If the motor time constant is less than 100 ms, or the ripple amplitudes of the DC-Link is too high, it's recommended to select higher switching frequencies.

Gear Ratio

A gear ratio between motor and driving shafts can be configured. This ratio defines the relationship between the motor shaft and driving output shaft. The position feedback and command will always be relative to the driving shaft.

$$\text{gear ratio} = \frac{\text{motor shaft revolutions}}{\text{driving shaft revolutions}}$$

The feature dual-loop cascaded position control can be used in such setups.

Advanced configurations

Motor Overload Protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating.

Cogging Torque Compensation Removes ripples caused by magnetic interaction between motor rotor and stator.

Field Weakening The velocity range of electric motors can be extended by weakening the magnetic field of the rotor linearly over the speed range.

Parameters

Object ID	Name	Description
0x2003	Motor Specific Settings	Parameters according to your motor specification
0x2010	Torque controller	Values Damping Ration and Settling time for tuning the current controller and switching frequency (PWM)
0x6075	Motor Rated Current	The RMS value of motor phase currents which can continuously pass through motor windings without

Object ID	Name	Description
		overheating the motor
0x6076	Motor Rated Torque	The amplitude of generated torque at the rated motor phase currents
0x6080	Max motor speed	The maximum possible speed of the Motor according to the data sheet
0x6091	Gear Ratio	Object for indicating the revolutions of motor and shaft

Generic brake output voltage

Overview

The voltage output of the brake can be controlled manually in real-time using the 'manual output voltage' mode of the brake release object.

> Note:
> The brake release object can be used for either controlling a brake or a generic voltage but not both.

This allows for custom brake release strategies on the master, as well as providing a generic output for a single phase device connected to this phase (e.g. a fan).

This voltage is applied at

- Phase D of SOMANET Node
- SBC out of SOMANET Node Safety

> Attention:
> The brake output can be **switched off safely** when SOMANET Node Safety is used but **not controlled safely**.

Usage

- Connect the device to the dedicated brake output of the drive and to Ground (GND).
- Set object ‘Brake options: Release strategy’ (subitem 0x2004:4) to ‘Manual output voltage’ (0), it is recommended to do so in ‘Brake Configuration’ of the Setup Wizard of OBLAC Drives.
- Use the SDO object ‘Brake options: Brake status’ (subitem 0x2004:7) for switching the output on and off:

  1: off

  2: on
- In the master assign ‘Brake options: Output voltage’ (subitem 0x2004:10) to PDO for controlling the output voltage in real-time, the unit is given in mV.
- While the master is running, control the output voltage by writing the value in mV to ‘Output voltage’.
- To turn off the output, write a 0 to ‘Output voltage’

> Note:
>
> The SBC signal has to be provided as well when the voltage shall be switched on with Node Safety.

Limitations

- Please note that the brake phase voltage control comes with limited accuracy.
- Use this feature only in low precision applications (e.g. load-dependant speed of a fan).

Parameters

Object ID	Name	Description
0x2004	Brake Release	• Subitem 0x2004:4 (Release strategy) can be used to activate the voltage on the brake output • Subitem 0x2004:7 (Brake status) turns it On and Off • Subitem 0x2004:10 (Output voltage) sets the voltage.

Brake configuration

In some cases, the motor is equipped with a mechanical brake which can be released by applying electrical voltage to the brake terminals.

Detailed description

The following diagram shows the steps for the release procedure:

A configurable voltage is applied to the positive terminal of motor brake (It is assumed that the negative terminal of the brake is connected to Ground).

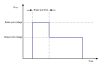


Image 1 - Timing diagram

When the motion system has entered operation enabled, the specified “Brake pull voltage” will be automatically applied for a configurable amount of time designated as “Brake pull time”. After this time period, the second voltage level “Brake hold voltage” will be applied to the brake terminals permanently. The brake hold voltage can be set to a lower value to save energy.

> Attention:
>
> Don’t set the hold voltage too low or you risk entering a marginally stable situation where any jerk can engage the brake.

Once the motion system leaves the state “operation enabled” the voltage is automatically dropped to Ground after the designated time “Brake engage delay” and the brake is mechanically activated.

Pin-brake strategy

A pin-brake is a special type of brake commonly used in light-weight robot joints. The brake usually consists of a solenoid linear actuator fixed to robot body and a notched wheel fixed to the rotating motor shaft.

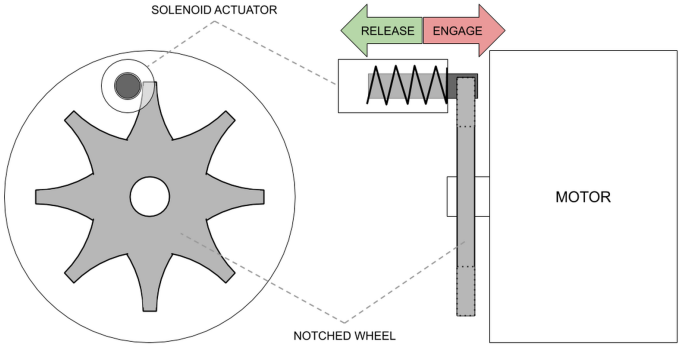


Image 2 - Pin brake (illustration exemplary)

To release a pin-type brake, the motor must move slightly so that the pin can slide out. This is mainly required with hanging loads (e.g. robot arm) where gravity is producing a torque pressing the wheel notch against the actuator pin. The following method is implemented to ensure reliable pin-type brake operation.

Releasing the Pin brake in Different OP Modes

Diagnostic Mode

In Diagnostics Mode, the brake is controlled by the master itself using the `Brake Status` object (0x2004:7). So putting the drive in OP ENABLED state in diagnostics mode will not start the Pin brake release process. Brake will remain in the engaged position. But if the drive has a relative encoder with Index and QEI index is not known then enabling the drive in diagnostics mode will only internally trigger the Temporary offset detection. Brake will remain in the engaged position and no process to release the Pin brake will be initiated.

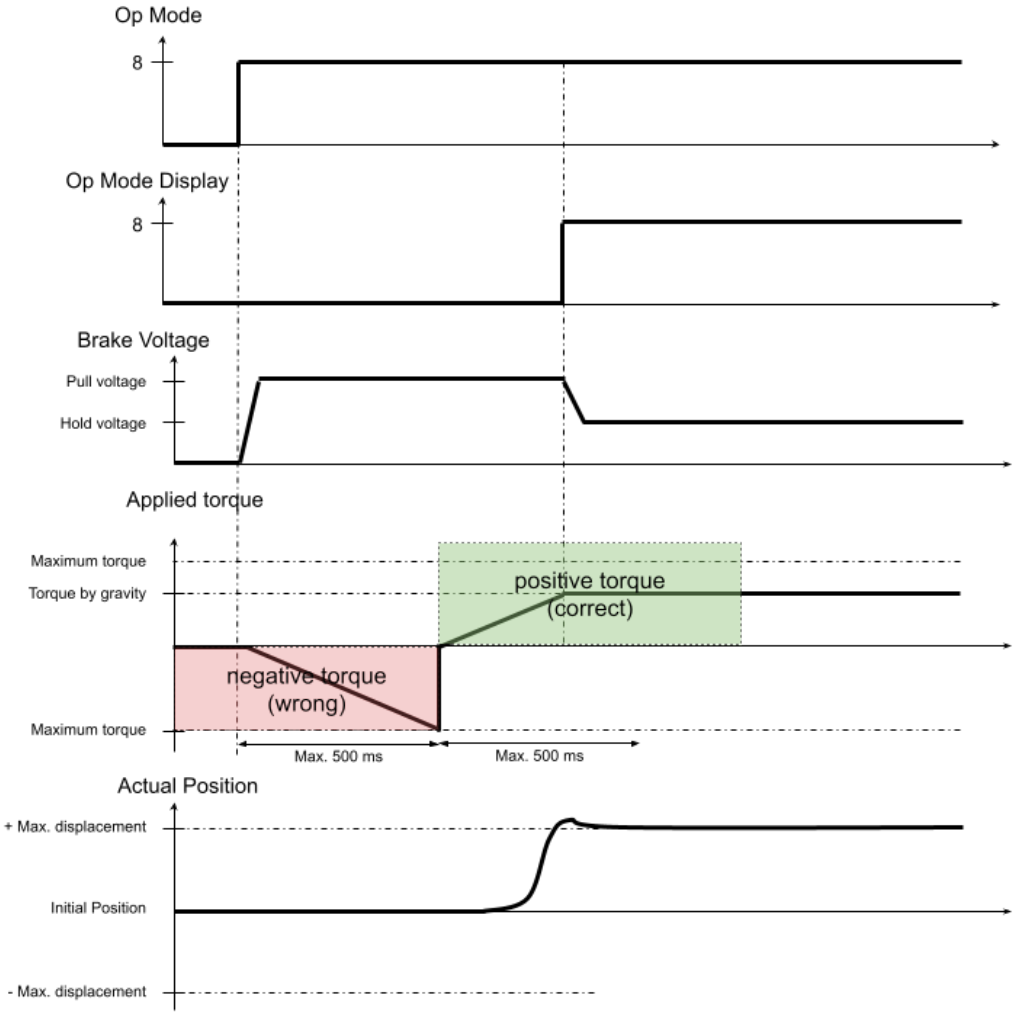
After the drive has been enabled, changing the brake status object to `DISENGAGE` will start releasing the pin brake.

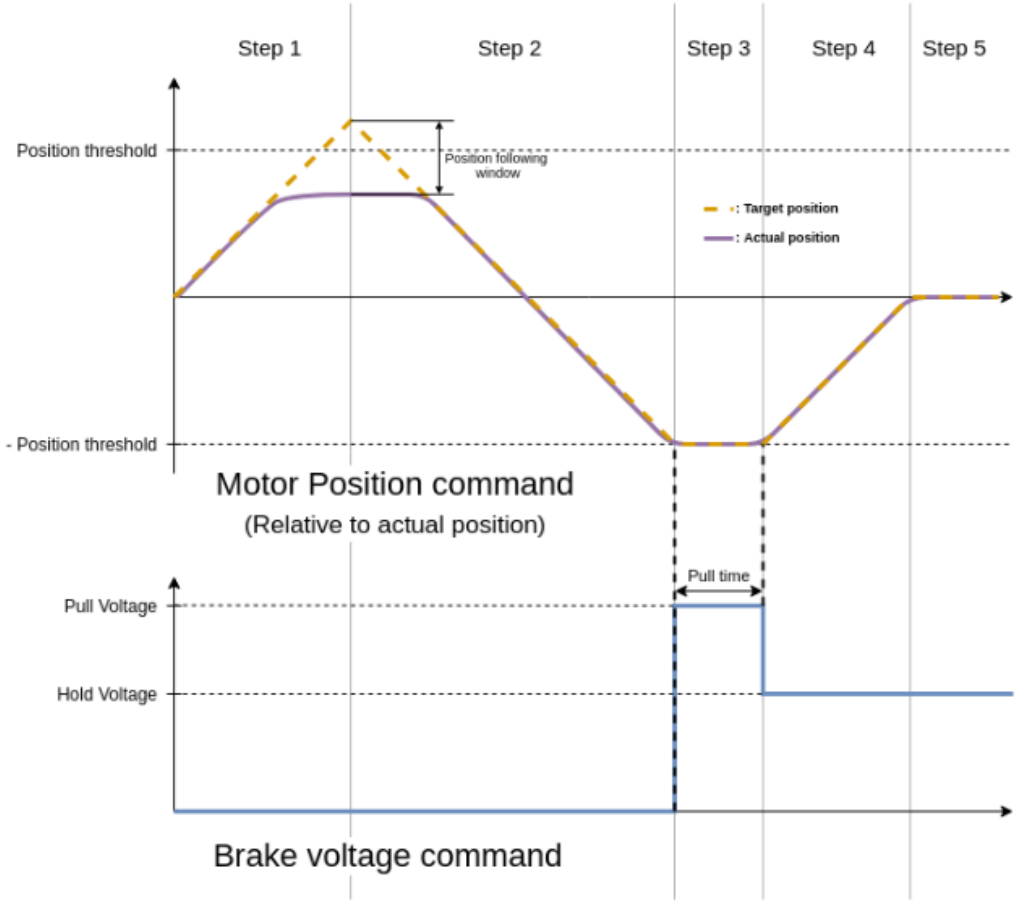
Manual Release using Brake Status Object

Controlling the brake using the brake status object 0x2004:7 will trigger the brake release procedure only if the drive is in Diagnostic mode and OP ENABLED state.

If the drive is not in OP ENABLED state, then releasing the brake will only apply voltage on PIN (Phase D).

Timing diagram





Step 0

In this step, the variables related to the pin brake release are initialized. The motion polarity is also determined based on the phase's connection (object 0x2003:05). The open loop profiler is configured.

Step 1

The open loop profiler moves the motor in positive / negative direction, depending on the polarity.

Step 2

The motor´s movement is blocked in any direction and it starts moving in the opposite direction.

Step 3

If the motor has reached the target position, the pull voltage is applied for the duration specified by pull time. During the application of pull time, the motor does not move.

Step 4

Target position is set to the initial position of the rotor and the open loop profiler motor moves to the initial position.

Step 5

Control is enabled, the drive accepts Position / Velocity / Torque targets.

The brake controller raises the torque progressively until it reaches the maximum displacement threshold. If it reaches the maximum torque threshold first, it rises the torque again in the opposite direction. When the torque is high enough to compensate for gravity, the load is lifted until the position reaches the threshold value, the torque stays to a holding torque value and the brake will be disengaged. After the release delay, the brake voltage is driven to “Brake hold voltage” and Op Mode display is set to Op Mode. The controller is then enabled. To use the pin-brake mode please select the correct “Release strategy” in the brake configuration section in OBLAC Drives commissioning software, or via direct object dictionary access.

> Note:
>
> The brake release is an open loop system. After the brake pull time it is assumed that releasing was successful and the hold voltage is applied.

Brake PWM switching frequency

The PWM switching frequency of the brake can be changed in object 0x2004:11 Brake options: Switching frequency.

- Enter the value 0 for 16 kHz
- Enter the value 1 for 32 kHz
- Enter the value 2 for 64 kHz

Custom brake modes

It is also possible to implement custom brake modes. For information please refer to Generic brake output voltage.

Parameters

Object ID	Name	Description
0x2004	Brake Release	All options that are required for brake release.

User defined velocity units

Overview

The *default unit* for all velocity and speed data is **rpm**

The *default unit* for all acceleration and deceleration data is **rpm/s**

In the case that the application requires a higher precision in velocity, the unit can be configured by the user through object 0x60A9 SI unit velocity.

Supported values are:

- 1 rpm (default value)
- 0.1 rpm
- 0.01 rpm
- 0.001 rpm

> Attention:
> As soon as this object is changed, **all related objects and subitems** also use the chosen unit and **must be scaled accordingly**. Unexpected behavior and accidents can occur if the chosen unit is not taken into account. For more information please refer to Objects affected.

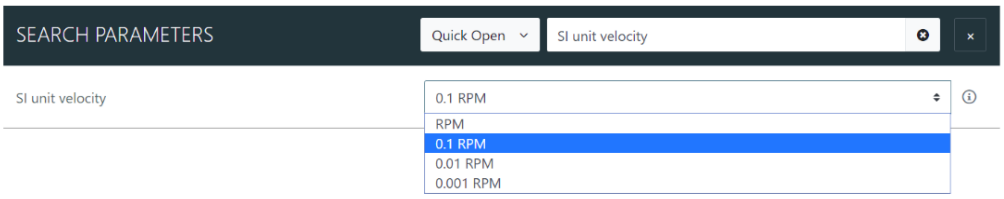
> Attention:
> The velocity reference frame is reported on the moving shaft. The target and actual velocity are scaled with a factor of gear-ratio. For better precision, it is recommended to change the SI unit to mrpm.

Usage

To configure the velocity unit write the following values through SDO to object 0x60A9 SI unit velocity:

- 1 rpm (value 0x00B44700 or 0d11814656) - default
- 0.1 rpm (value 0xFFB44700 or 0d4290004736)
- 0.01 rpm (value 0xFEB44700 or 0d4273227520)
- 0.001 rpm (value 0xFDB44700 or 0d4256450304)

Since version 19.1.0 of OBLAC Drives, this can be done directly in the UI using Quick Open:



After changing the unit please change all speeds /acceleration objects according to the new precision (see section below for details).

Click on “save to device” to keep the configuration.

Objects affected

- 0x606B Velocity demand value
- 0x606C Velocity actual value
- 0x606D Velocity window
- 0x606F Velocity threshold
- 0x6080 Max motor speed
- 0x6081 Profile velocity
- 0x6083 Profile acceleration
- 0x6084 Profile deceleration
- 0x6085 Quick stop deceleration
- 0x609A Homing acceleration
- 0x60B1 Velocity offset
- 0x60FA Control effort
- 0x60FF Target velocity

These subitems are also affected by the value:

- 0x6099:1 Homing speeds: Speed during search for switch
- 0x6099:2 Homing speeds: Speed during search for zero
- 0x2010:6 Torque controller: Field weakening starting speed
- 0x2010:7 Torque controller: Field weakening ending speed
- 0x2012:4 Position controller: Position loop integral limit

- 0x2013:6 Gain scheduling: Position loop integral limit 0
- 0x2013:15 Gain scheduling: Position loop integral limit 1
- 0x2013:2 Gain scheduling: Threshold velocity 0
- 0x2013:11 Gain scheduling: Threshold velocity 1

Example

When a dual-sensor setup is used with one encoder mounted after a reducer with a gear ratio of 1:100 the default velocity unit rpm causes severe inaccuracy in the motion control.

1. In OBLAC Box choose 0.001 RPM for SI unit velocity.

   When using a version of OBLAC Box before 19.1.0, manually write 0d4256450304 to object 0x60A9 SI unit velocity through SDO. The terminal of OBLAC Box can be used for this operation with the following code line:

   ```
   ethercat download -p0 0x60A9 4256450304
   ```

2. OBLAC Drives will automatically scale the affected objects.

Parameters

Object ID	Name	Description
0x60A9	SI unit velocity	This object defines the velocity unit.
0x606B	Velocity demand value	Output of position controller in position mode/trajectory generator in velocity mode.
0x606C	Velocity actual value	Provides the value read at the velocity sensor (or the derivation of the position signal).
0x606D	Velocity window	This value reflects each side of the target velocity.
0x606F	Velocity threshold	Configures the velocity that is considered zero velocity.
0x6080	Max motor speed	Sets the maximum allowed speed for the motor in either direction.

Object ID	Name	Description
0x6081	Profile velocity	Configures the velocity target for the end of the acceleration ramp during profiled motion.
0x6083	Profile acceleration	Configures the acceleration used during profiled motion.
0x6084	Profile deceleration	Configures the deceleration used during profiled motion.
0x6085	Quick stop deceleration	Deceleration used for stopping the motor when quick stop function is activated.
0x609A	Homing acceleration	Configures the acceleration and deceleration to be used during a homing operation.
0x60B1	Velocity offset	Sets a feed-forward velocity as input to the velocity controller.
0x60FA	Control effort	Provides a measure of the control effort (the output of the position control loop).
0x60FF	Target velocity	Sets the target velocity and will be used as input to the trajectory generator.
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)
0x2013	Gain Scheduling	Parameters needed for Gain Scheduling

Commutation Offset Detection

Overview

Controlling a motor with Field Oriented Control requires precise measurement of the motor's electrical angle. With phase order and commutation angle offset detection (hereinafter referred to as

“Offset Detection”), the relationship between the motor electrical angle and the motor mechanical angle (position sensor feedback) is identified, afterwards the motor position sensor feedback is used to determine the motor’s electrical angle for motion control.

> Note:
>
> This feature is needed for every motor control setup. The application will not work properly before the Offset Detection routine is successfully executed.
>
> If an incremental encoder without index is used for commutation, the offset detection has to be done on every power on. Otherwise, the routine only needs to be done once during the first commissioning and after mechanical changes of the motor or of the commutation encoder.

Basic Principles

The Offset Detection routines identify two parameters for the relationship between the motor electrical angle (rotor magnetic field angle) and the motor position sensor feedback:

1. The motor phase configuration (NORMAL or INVERTED) (0x2003:5)

   Normally, both rotor angle and sensor angle should increase/decrease at the same time. This situation is called “NORMAL motor phase configuration” and is shown in Image 1.

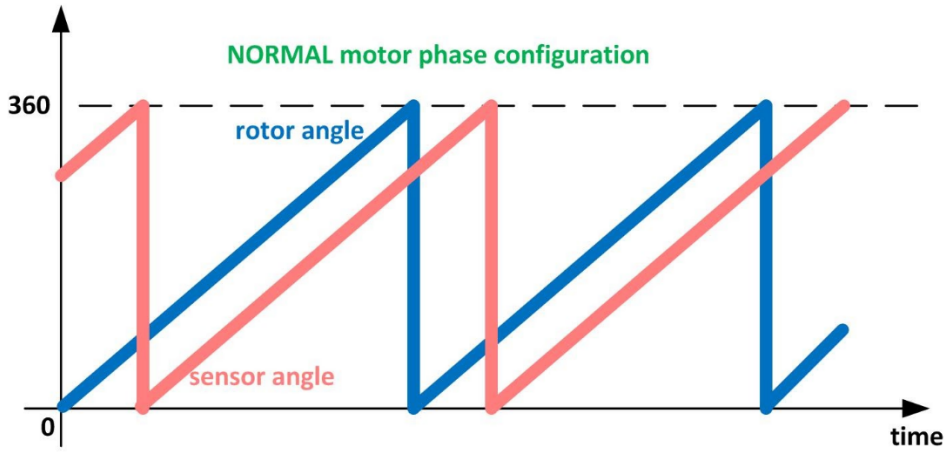


Image 1 - Normal motor phase configuration (sensor angle and rotor angle are both increasing/decreasing simultaneously)

However, sometimes the angle of position sensor is not increasing in the same direction of rotor angle. This can happen if the position sensor is mechanically flipped, or if two of the stator phases are flipped. Image 2 shows such a situation where the rotor angle and the position angle are not increasing/decreasing in the same direction. This situation is called *Inverted motor phase configuration*.

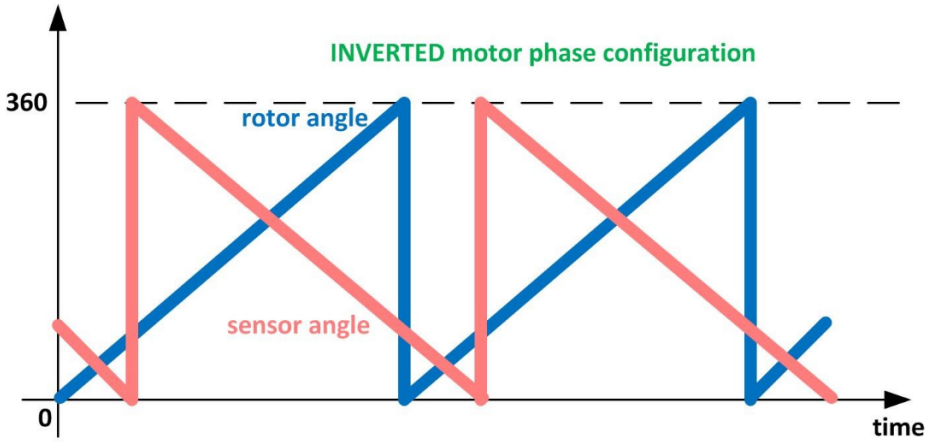


Image 2 - Rotor angle and position sensor angle are not increasing/decreasing in the same direction (Inverted motor phases)

2. The commutation angle offset (0x2001)

   In most cases, the provided sensor angle contains a shift respect rotor angle. Image 3 shows such a case where the provided sensor angle has an offset respect rotor angle:

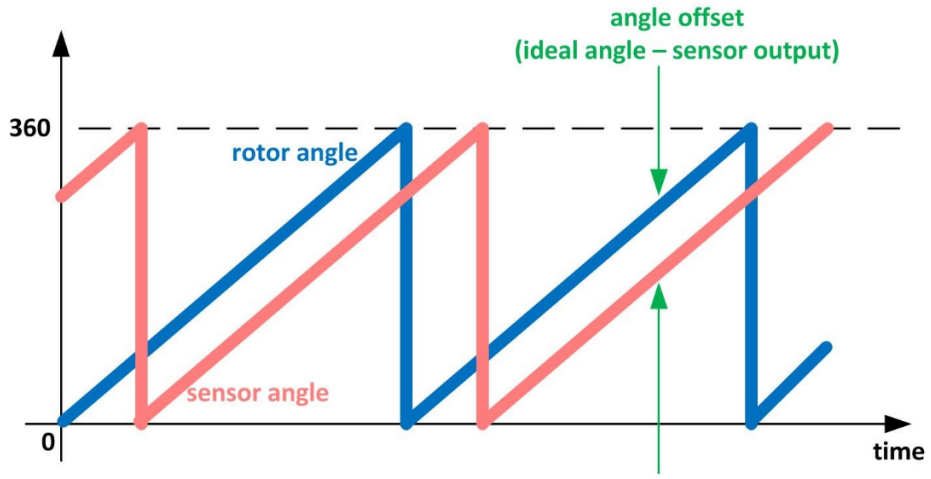


Image 3 - In most cases, the provided sensor angle has a commutation angle offset

Offset Detection is carried out in the Diagnostics Opmode using OS-Commands.

Note:

When quick stop is commanded during Offset detection, the procedure is aborted and the drive is disabled.

Requirements

Load requirements

- The active load of the system should be lower than motor rated torque
- The position feedback sensor on the motor side should be configured properly and provide proper feedback of the motor position
- The motor should be configured correctly
- The mechanical system should not block the motor
- The brake should be configured correctly and work properly during the routine so that it doesn’t block the system
- The power supply used should be able to provide the voltage / current required by the routine

Note:

If the requirements are not met, the Offset Detection routine fails and the detection result can be wrong, leading to further failures.

Note:

The precision of the commutation angle offset detection depends on the resolution of the encoder that is used. For example, if HALL sensors with a resolution equal to 60 degrees are used, the precision of commutation angle offset detection is expected to be 60 degrees as well.

Autophasing methods

Three different methods for finding the “commutation angle offset” are implemented. The method can be selected manually (in subitem 0x2009:3). The optimal method is determined by the setup and the use case.

> Note:
>
> Large single side loads during the routine (such as the gravitational load) can lead to lower detection accuracy with Offset Detection method 0 and 1.

Method 0

Method 0 is the option used by default. This algorithm has the following characteristics:

- It needs no tuning
- It is able to hold the load
- The offset can be precisely measured
- It will rotate the rotor up to 100% of a pole-pair range

Method 1

> Attention:
>
> This method is still in prototype phase and can be tested at the user’s own risk.

This algorithm has the following characteristics:

- It needs previous tuning of the controller with a dedicated set of gains: the phasing controller whose parameters are located in object 0x2009
- The offset can be found with a limited movement range. With a well tuned PID controller the movement is typically less than 5 degrees
- The offset can be measured very precisely

Method 2

> Attention:
>
> This method is still in prototype phase and can be tested at the user's own risk.

This algorithm has the following characteristics:

- It needs no movement (no rotation)
- If a brake is available, the brake should be engaged while offset detection method 2 is in progress.

> Attention:
>
> If no brake is available and mechanical load is relatively small (compared to rated motor torque), then offset detection method 2 can lead to rotor movement

- It is not able to hold the load. As a result, **when a brake is present, the brake will remain engaged during the procedure of offset detection method 2.**
- It can take up to 250 milliseconds.
- The precision of method 2 is not as high as methods 0 or 1.

> Note:
>
> This method can cause audible noise when it's executed.

Comparison of methods

The optimal method depends on the use-case. The following tables shows the advantages and disadvantages of each method.

Method #	Tuning necessary?	Able to hold a load?	Precision of measurement	Rotation required	Notes
0	No	Yes	Good	up to 1 pole-pair range	
1	Yes	Yes	Good	Depends on quality of tuning. Typically less than 20 elec-trical degrees.	
2	No	No*	Less good	None	Duration up to 250 ms Can cause audible noise

* This procedure is usually carried out with applied brakes.

Common Use Cases

Motion is allowed

- Method 0/1 should be used
- Method 0 gives perfect precision without any previous tuning.
- If the final system has coupled joints or heavy or single side loads it might be required to perform Offset detection before assembly for optimal performance.

- If the movement of up to one pole pair is not acceptable method 1 can be selected, where the movement can be much less. The PID controller used for method 1 needs to be tuned carefully.

> Note:
> Try moving the arm to a position in which gravity has no or little impact, for optimal performance of the detection. For example the center of gravity should be right above or right below the axis of motion.

Motion is not allowed

When no movement is allowed during commissioning, method 2 can be used. This may happen when the setup is already in the application. Note that the method has less detection accuracy, thus the commutation efficiency can be less. In this case, it's recommended to redo Offset Detection when the setup is allowed to move.

Special case: Offset detection Method 2 with an Incremental encoder with index

If an ABI encoder is used for commutation, a routine called "finding index" is usually required when enabling the drive (the first time after booting the system), which will rotate the motor to find the index of the sensor. With commutation method 2, the drive can find an internal temporary offset and use this offset until the index is detected. Once the index is detected, it replaces the temporary offset by its corresponding value in the object dictionary. In practice, Offset Detection can be carried out with method 0/1 during commissioning, the result can be saved and method 2 can be executed. This way, no movement is required when enabling the setup, and the torque control performance of the setup is still promised.

Usage

Offset detection is usually carried out in the setup section of OBLAC Box. The commutation offset detection method is chosen with subitem 0x2009:3. The default method is 1.

> Attention:
> When the existing Offset Detection results become invalid, try locating the cause and fix the system-level failure instead of running Offset Detection again. Else further failures can happen.

The procedure can also be triggered manually for production use:

Steps to activate the “commutation angle offset chain”:

- In object 0x6060: (Modes of operation) set the value to -2 (Diagnostics mode)
- In subitem 0x2009:3 select the offset detection method (values: 0, 1 or 2)
- In object 0x6040 Controlword switch the CiA402 status to “Operation_Enabled”
- The procedure will start automatically.
- Trigger the following OS commands to carry out offset detection:
    - Command 6: Open phase detection
    - Command 8: Phase resistance measurement
    - Command 9: Phase inductance measurement
    - Command 7: Pole pair detection
    - Command 4: Motor phase order detection
    - Command 5: Commutation offset measurement

Autophasing during initial configuration of a drive with absolute encoder:

The following steps are mandatory:

- Command 6: Open phase detection
- Command 7: Pole pair detection (on new type of motor)
- Command 4: Motor phase order detection
- Command 5: Commutation offset measurement

Autophasing after power on with incremental encoder:

The following steps are mandatory:

- Command 4: Motor phase order detection
- Command 5: Commutation offset measurement

> Note:
>
> Motor phase order detection is mandatory in order not to damage the system.

Tuning gains (method 1)

The gains for PID tuning can be found in object 0x2009:

- Subitem 0x2009:4 defines K_p for method 1.
- Subitem 0x2009:5 defines K_i for method 1.
- Subitem 0x2009:6 defines K_d for method 1.

> Note:
>
> As a rule of thumb, the following approach can be applied:
>
> Start values of the gains:
>
> $K_p = 1$
>
> $K_i = 1$
>
> $K_d = 0$
>
> Increasing and decreasing K_p and K_i will have the same effect as increasing them in a simple PID position controller.
>
> If needed, the increase in K_d should be done in very small steps (such as changing from 0 to 0.001), this will reduce the oscillations of the load.

Parameters

Object ID	Name	Description
0x2001	Commutation angle offset	Represents the required shift for position sensor output.
0x2003	Motor Specific Settings	Details for your motor specification.
0x2009	Commutation offset	• Subitem 0x2009:1 shows the state of the commutation offset. • Subitem 0x2009:2 defines the amount of torque that is applied during the offset detection process.

Object ID	Name	Description
		• Subitem 0x2009:3 determines the method to be used for offset detection. • Subitem 0x2009:4 defines K_p for method 1. • Subitem 0x2009:5 defines K_i for method 1. • Subitem 0x2009:6 defines K_d for method 1.
0x6075	Motor Rated Current	Determines what percent of rated motor currents should be injected into the motor phases during the offset detection procedure.
0x6076	Motor Rated Torque	The rated torque of your Motor according to the data sheet.
0x6060	Modes of operation	Selects the mode of operation.
0x6040	Controlword	Sets the status of the drive

Power Limit

Overview

In case the DC power supply of the Drive has limited output power (for example limited output I_{DC} at a certain V_{DC} level) this feature can be used to make sure that the output power drawn from the power supply is limited.

Use case:

Battery powered system

A good example of such cases are applications where batteries are used, and their output currents should remain below a certain limit.

This feature does not measure the mechanical torque directly. Instead it is using the “computed electrical torque” and assumes that the “computed electrical torque” is equal to the real generated mechanical torque of the motor.

It calculates the rotational speed of the motor with values from the position sensor at the motor shaft.

Requirements

This feature works in CST, CSV, CSP, PT, PV and PP modes.

The following parameters must be properly set from the motor data sheet:

- Motor torque constant
- Position sensor configurations
- Number of pole pairs

Detailed Description

Mechanical output power of the Drive in any condition is equal to:

$$P_{out}\left[W\right] = \tau\left[Nm\right] \cdot \omega\left[\frac{rad}{s}\right]$$

where τ represents Torque and ω represents the rotational speed.

At any operational point, the Drive has access to rotational speed. As a result the limit of generated torque can be calculated as:

$$\tau_{limit}\left[Nm\right] = \frac{P_{out,max}\left[W\right]}{\omega\left[\frac{rad}{s}\right]}$$

This limit is updated in each execution loop of the torque control, and it limits the absolute value of reference torque.

> Note:
> By default the power of each Drive is internally limited to 110% of its maximum peak output power. As a result, this object will only be effective if its value is less than 110% of Drive’s maximum peak output power.

Parameters

Object ID	Name	Description
0x200B	Max power	Sets the limit in Watt
0x203F	Error Report Object	Shows the warning “MxPwrLmt” once the limit is reached

Encoders and I/O

Besides various standard interfaces for absolute and incremental encoders, SOMANET devices are equipped with a multitude of general purpose digital and analog IOs. These can be both directly propagated to a higher level controller via the field-bus interface, or configured to internally be used for triggering, homing, etc.

It’s also possible to connect devices such as valves, magnetic actuators or LEDs to digital outputs.

For attaching a brake, please refer to brake configuration.

Analog In 749
Analog Input Scaling 751
Digital I/O 754
Digital GPIO 755
Encoder Configuration 766
Encoder sampling frequency 768
Commutation with Hall and ABI combined 769
Input counter 770
Addressable LED control 771

Analog In

Analog sensors usually output a voltage or a resistance that can be interpreted by the firmware. When the output is a voltage, the signal can be directly read by the analog input, but if it’s a resistance (e.g. resistance temperature detectors), a voltage divider must be used to generate a voltage equivalent to the resistance.

> Important:
>
> SOMANET Node does not incorporate a voltage divider itself, you must attach one separately.

SOMANET servo drives come with a default configuration for the analog inputs that can be changed on request.

For details please refer to the Analog In specification .

Using single-ended sensors on differential inputs

A single-ended encoder can be attached to the differential ports, if the voltage range fits only to the differential inputs. The signal must be connected to the +pin of the Analog In and the Ground to the -pin.

> Note:
>
> In this case, half of the analog input range will be lost.

Nonlinear conversion

- Analog Input Scaling: Nonlinear conversion of an analog voltage using a 5th order polynomial function.

Parameters

Object ID	Name	Description
0x2401	Analog input 1	Parameter holding the value of analog input 1
0x2402	Analog input 2	Parameter holding the value of analog input 2
0x2403	Analog input 3	Parameter holding the value of analog input 3
0x2404	Analog input 4	Parameter holding the value of analog input 4

Analog Input Scaling

The analog input value is converted into digital ticks from 0 (representing U_{min} of the selected input, e.g. -5 V) to 65535 (representing U_{max} of the selected input, e.g. +5 V) which must be computed externally for further use.

This feature allows a nonlinear conversion of one of the available analog inputs using a polynomial function up to 5th order and to read the value directly.

It's also possible to describe a voltage divider connected to the analog input for sensors that output a resistance as measurement value.

> Use case:
>
> Temperature sensor: Applications include converting the output of a temperature sensor directly into degrees Celsius, Fahrenheit or any other unit desired by the user.
>
> For details about using temperature sensors check the chapter Temperature sensors in the System integration guide.

> Use case:
>
> Torque sensor: Applications include converting the output of a torque sensor directly into mNm, Nm or any other unit desired by the user.

Requirements

- The hardware must support at least one analog input

Usage

Wiring

Follow this wiring scheme when attaching an external voltage divider.

Software configuration

The object used for this feature is 0x2038 (External scaled measurement). The converted value will be reflected in Subitem 0x2038:1.

The polynomial used for the conversion can be described with the parameters "Constant a0" (0x2038:4) to "Constant a5" (0x2038:9), which are the coefficients for the polynomial function.

The analog input to be used can be defined using the parameter “Analog input” in subitem 0x2038:2.

This parameter has 4 possible options, which are the standard analog inputs (values: 1-4).

- Analog input selection k (0x2038:2): Select the scaled measurement source. Values:
  - 1 = Analog Input 1
  - 2 = Analog Input 2
  - 3 = Analog Input 3
  - 4 = Analog Input 4

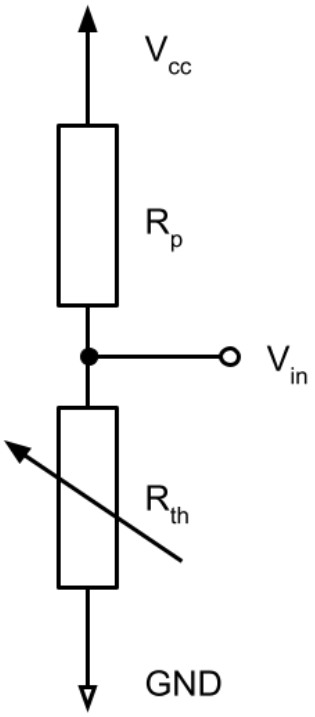
The unscaled raw analog input values are filtered through a first order low pass filter before they are scaled. The cut-off frequency of this low-pass filter can be configured in subitem 0x2038:10.

> Note:
>
> The external scaled measurement feature is using the filtered analog input value. Therefore all the analog inputs mentioned in this section are referring to the filtered analog input.

A 5th order polynomial function with or without voltage-divider is supported.

This feature is also designed to have a different behavior depending on the value of the parameter “Resistance” (0x2038:3). If this value is set to 0, the polynomial will be directly applied to the value of the ADC count, which has a range of 0-65535 (0 to V_{cc}).

If a voltage divider is connected to the analog input, as shown in the figure below, the value of the “Resistance” parameter (0x2038:3) defines the value of the pull-up resistor R_p of this voltage divider.



- V_{cc} represents the value of voltage at which the ADC is connected to, causing the ADC to count an output of 65535.
- R_p represents the pull-up resistor.
- R_{th} is a resistive sensor, e.g. a thermistor or strain gauge.
- V_{in} is the voltage that is connected to the selected analog input.

The cut-off frequency of the low-pass filter (0x2038:10) for the analog input can also be tweaked to further fine-tune the unscaled analog input before feeding it to the scaling algorithm.

Thresholds

Upper and lower level thresholds can be configured in subitems 0x2038:11 and 0x2038:12. When these thresholds are violated, the error **ExAnSnsr** is triggered in the Error report object.

> Note:
>
> When the value approaches 95% of the defined thresholds, a warning is triggered.

Polynomial

If R_p := 0 (no voltage divider)

$$y(u_k) = a_5 * u_k^5 + a_4 * u_k^4 + a_3 * u_k^3 + a_2 * u_k^2 + a_1 * u_k + a_0$$

If $R_p > 0$ (user specified pull-up resistor for voltage divider)

$$y(u_k) = a_5 * R_{th}^5 + a_4 * R_{th}^4 + a_3 * R_{th}^3 + a_2 * R_{th}^2 + a_1 * R_{th} + a_0$$

with

$$R_{th} = \frac{u_k}{4096} * \frac{R_P}{1 - \frac{u_k}{4096}}$$

The coefficients and the pull-up resistor values are specified in object 0x2038, subitems 2-9

- 0x2038:3 (Resistance R): Select the pull-up resistor connected to the selected analog input. The value is given in Ohm.
- 0x2038:4 (Constant a0): Constant a0 of the polynomial (Offset).
- 0x2038:5 (Constant a1): Constant a1 of the polynomial (1st order coefficient)
- 0x2038:6 (Constant a2): Constant a2 of the polynomial (2nd order coefficient)
- 0x2038:7 (Constant a3): Constant a3 of the polynomial (3rd order coefficient)
- 0x2038:8 (Constant a4): Constant a4 of the polynomial (4th order coefficient)
- 0x2038:9 (Constant a5): Constant a5 of the polynomial (5th order coefficient)

Parameters

Object ID	Name	Description
0x2038	External scaled measurement	Transforms an analog input to an output by a 5th order polynomial

Digital I/O

SOMANET supports several Digital I/O which can be either used as GPIO or SPI SSI (single-ended), I²C, BiSS (single-ended).

Details and default settings are described in detail for Encoders and IO.

Configuration of Digital I/Os as Input or Output

Most Digital I/Os are configurable as either input or output independently if they are used as general purpose input/output (GPIO). This is done in 0x2210 GPIO pin configuration.

Subitems 1-4 describe the available pins and can be configured with these values:

- 0 Disabled (default)
- 1 Input without pull-down
- 2 Input active high - a pull-down resistor will be activated
- 3 Output

Using Digital I/Os for input actions or output events

It's possible to use the Digital Inputs of the Drive to command actions and to configure the Digital Outputs to trigger when certain events happen. This can be used to control some aspects of the Drive with an external signal or to control an external device with the Drive:

- Digital GPIO: Assigns actions to changes in the Digital Inputs and triggers Digital Outputs

Parameters

Object ID	Name	Description
0x2210	GPIO	Select if Digital I/Os are input or output

Digital GPIO

Overview

This feature allows using the Digital Inputs of the Drive to command actions and to configure the Digital Outputs to trigger when certain events happen. This can be used to control some aspects of the Drive with an external signal or to control an external device with the Drive.

Available features are:

Value	Definition	
0		Disabled
2	[IN]	General Purpose Input
3	[OUT]	General Purpose Output
4	[IN]	Home Switch Active High - Level Triggered
5	[IN]	Home Switch Active Low - Level Triggered

Value	Definition	
6	[IN]	Positive Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)
7	[IN]	Positive Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)
8	[IN]	Negative Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
9	[IN]	Negative Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
10	[IN]	Interlock (Disable Operation) Active Low - Level Triggered
11	[IN]	Fault Reset - Edge Triggered
12	[IN]	Interlock (Fault Reaction) Active Low - Level Triggered
13	[IN]	Time Stamped Rising Edge - Edge Triggered
14	[OUT]	Mimics Target Reached (tr) Bit of Statusword
15	[OUT]	Mimics Internal Limit Active (ila) Bit of Statusword
16	[OUT]	Timestamped Triggered Output
17	[OUT]	Position Triggered Output (Pulse)
18	[OUT]	Mimics Fault (f) Bit of Statusword

Value	Definition	
19	[OUT]	Mimics Switched On (so) Bit of Statusword
20	[OUT]	Mimics Ready to Switch On (rtso) Bit of Statusword
21	[OUT]	Mimics Operation Enabled (oe) Bit of Statusword
22	[OUT]	Mimics Voltage Enabled (ve) Bit of Statusword
23	[OUT]	Mimics Switch On Disabled (sod) Bit of Statusword
24	[OUT]	Mimics Quick Stop (qs) Bit of Statusword
25	[OUT]	Mimics Warning (w) Bit of Statusword
26	[OUT]	External LED Control

Usage

You can access the features in OBLAC Drives: Enter Feedback & IO/Digital IO

Manual configuration

The actions and events on GPIO pins can also be configured manually in object Object 0x2210 (GPIO Pin Configuration).

In this object a feature can be selected for each Digital I/O pin.

Some features require additional settings in other objects as well.

> Attention:
>
> Ensure that the selected pin allows the direction of the feature:
>
> An **input** action can only be configured on a Digital **In** pin.
>
> An **output** event can only be configured on a Digital **Out** pin.

Attention:

Each feature can only be used on one pin (except for General Purpose Input and General Purpose Output)

Description of features

General purpose In- and Output (GPIO)

General Purpose Input (2)

The high and low logic (voltage levels) at the selected GPIO pin will be reflected in the corresponding bits of object 0x60FD (Digital Inputs) as 1 and 0 respectively.

General Purpose Output (3)

The value of the selected digital output is high if the corresponding bit in sub-object 0x60FE:1 is high and the corresponding mask bit in sub-object 0x60FE:2 is set as 1.

The output value will be the value set in the object 0x60FE:1 (Digital Output).

Using Digital GPIO with Home switches and limit switches

Note:

Limit switch and Home switch configuration are used when the drive is in Homing Mode.

Limit switches can also trigger Quick Stop in all modes. In homing mode Quick Stop is triggered by the limit switch if it is not being used by the homing method.

Home Switch Active High - Level Triggered (4)

Active and inactive state of the Home switch will be reflected as 1 and 0 respectively at the selected digital input.

Home Switch Active Low - Level Triggered (5)

Active and inactive state of the Home switch will be reflected as 0 and 1 respectively at the selected digital input.

> Note:
> Irrespective of the switch type (Active High or Active Low), active and inactive states of the home switch will be shown as 1 and 0 respectively in bit 2 (home switch) of the Digital Input Object 0x60FD.

Positive Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct) (6)

Active and inactive state of the Positive limit switch will be reflected as 1 and 0 respectively at the selected digital input.

Positive Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct) (7)

Active and inactive state of the Positive limit switch will be reflected as 0 and 1 respectively at the selected digital input.

> Note:
> Irrespective of the switch type (Active High or Active Low), active and inactive states of the positive limit switch will be shown as 1 and 0 respectively in bit 1 (positive limit switch) of the Digital Input Object 0x60FD.

> Note:
> Positive Limit switch also triggers Quick Stop, if the drive is either in any opmode other than HOMING or the drive in is HOMING mode but the switch is not being used by the homing method. If Positive Limit Switch has been activated in this case, then bit 2 (qs) of the Controlword will be cleared and Quick Stop will be executed.

Negative Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct) (8)

Active and inactive state of the Negative limit switch will be reflected as 1 and 0 , in the corresponding bit (according to configured GPIO) of the object 0x60FD (Digital Inputs)

Negative Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct) (9)

Active and inactive state of the Negative limit switch will be reflected as 0 and 1 , in the corresponding bit (according to configured GPIO) of the object 0x60FD (Digital Inputs)

> Note:
> Irrespective of the switch type (Active High or Active Low) , active and inactive states of the negative limit switch will be shown as 1 and 0 respectively in bit 0 (negative limit switch) of the Digital Input Object 0x60FD.

> Note:
> Negative Limit switch also triggers Quick Stop, If drive is either in any opmode other than HOMING or drive in is HOMING mode but the switch is not being used by the homing method. If Negative Limit Switch has been activated in this case, then bit 2 (qs) of the Controlword will be cleared and Quick Stop will be executed.

Driving out of the limit switch is only possible in the opposite direction, the concrete behavior is mode-dependant:

Modes CSP, PP , PV , CSV , CST and PT

If a target is commanded which would move the axis further into the direction of the switch, the motor will stand still, since further movement in the direction of switch is not allowed.

Cogging torque compensation

If the switch is active and the user tries to switch to cogging torque compensation, this will not be allowed.

The error based on the activated switch will be raised by firmware.

Diagnostics mode

The following OS commands are not executed, if either of the limit switches is active:

- OS command 2 (Open Loop Field Mode)
- OS command 4 (Motor Phase Order Detection)
- OS command 5 (Commutation offset detection **with Methods 0 and 1**)
- OS command 6 (Open phase detection)
- OS command 7 (Pole pair detection)
- OS command 8 (Phase Resistance Measurement)
- OS command 9 (Phase Inductance Measurement)

If any of these OS commands is called while a limit switch is active, the response will be **OSCMD_ERROR_CMD_NOT_ALLOWED**.

The following OS commands can be executed, even if either of the limit switches is active:

- OS command 0 (Encoder register communication)
- OS command 1 (iC-MU calibration mode)
- OS command 3 (HRD streaming)
- OS command 5 (Commutation offset detection **with Method 2**)

Homing modes

If the switch is active then Homing is only allowed in those modes that will take the axis away from the activated switch. All those methods which try to move the axis further into the limit switch will raise an error.

The error raised depends on the limit switch being active:

- For a positive limit switch the error raised will be **PosLmAct**
- For a negative limit switch the error raised will be **NegLmAct**

Input actions

> Use case:
> Interlock: When connecting different devices for a common process it's often necessary to ensure the correct order of operation. This can be done by routing the signals to the master but it comes at a cost for reaction-time, data load and additional wiring. Input actions allow

nterlocking an operation directly at the Drive to guarantee the correct order of the process.

Interlock (Disable Operation) Active Low - Level Triggered (10)

Clear bit 3 (eo) of controlword when the selected digital input is low.

This will cause the drive to transition into Operation disable state.

Interlock will remain active as long as the configured GPIO remains low and transitioning into OP Enabled state will not be possible. Transitioning into OP Enabled state will only be possible when interlock is deactivated by setting the GPIO input high. Bit 3 (interlock bit) of the Digital Inputs Object 60FD will also be high when Interlock is active.

Fault reset (11)

Set bit 7 (fr) of controlword when the selected digital input is high.

Interlock (Fault Reaction) Active Low - Level Triggered (12)

Raises the fault “Interlck”, when the selected digital input is low.

This will cause the drive to transition into Fault Reaction State.

Interlock will remain active as long as the configured GPIO remains low and entering into OP Enabled state will not be possible. If OP Enabled state is attempted to be entered while interlock is active, *Interlck* fault will be raised again by the firmware. Transitioning into OP Enabled state will be possible when interlock is deactivated by setting the configured GPIO input high. Bit 3 (interlock bit) of the Digital Inputs Object 60FD will also be high when Interlock is active.

Time Stamped Rising Edge - Edge Triggered (13)

Report a rising edge timestamp with 2 µs accuracy if GPIO is a fast port (as indicated in the table below). Otherwise the accuracy is 62.5 µs. The value of this GPIO is updated in shared memory at 16 kHz.

Output events

Mimics Target Reached (tr) Bit of Statusword (14)

The value of the selected Digital Output mimics bit 10 of the Statusword (tr). When the “tr” bit is high, the configured digital output will be high

Mimics Internal Limit Active (ila) Bit of Statusword (15)

The value of the selected Digital Output mimics bit 11 of the Statusword (ila). When the “ila” bit is high, the digital output value will be high.

Timestamp triggered Output (16)

Trigger the Digital Output at a certain timestamp. The output value will be the value set in the object 0x60FE:1 (Digital Output).

The value specified in subitem 1 of object 0x2211 (GPIO output events) is the Timestamp at which the GPIO should trigger. The unit is in 10 ns.

The value in 0x2211:1 can be used for setting this timestamp. But the unit must be converted to 10 of ns.

> Note:
>
> The value need to be converted to be in the same time reference (1 µs vs. 10 ns)

> Use case:
>
> The value read in object 0x20F0:0 is 2385309 µs. The Timestamp triggered output shall be triggered 15000 µs after the current time. Both times need to be converted to the units of 0x2211:1 and added: 2385309 * 100 + 15000 * 100 = 240030900.

> Attention:
>
> When the DC Clock feature is not used, there is a drift between the timestamp and the DC Clock.

> Note:
>
> If the GPIO is a fast port, this is activated with an accuracy of 10 ns. Otherwise the accuracy is 62.5 µs.

Please refer to the table below for the type of port:

	type of port
Digital I/O 1	Fast port
Digital I/O 2	Fast port
Digital I/O 3	Fast port
Digital Output 4	N/A
Digital Input 7	N/A

Position Triggered Output (Pulse) (17)

Output a configurable width pulse to the selected Digital Output at a certain encoder position. The encoder port that is used can be selected in subitem 2 of object 0x2213:

	encoder port
Encoder Connector 1	Encoder Port 1
Encoder Connector 2	Encoder Port 2
Encoder Connector 3	Encoder Port 3 / Digital IO
Encoder Connector 4	
Encoder Connector 5	

> Note:
> If an ABI encoder is used, the response will be almost immediate (within 0.34 µs). If another encoder type is selected, the response will be slower (within 62.5 µs).

Object 0x2213 allows further configuration of this output event by setting the following parameters:

- Pulse Width
- Position Setpoint at which the pulse needs to occur at GPIO
- Selection of the encoder for which setpoint is defined.

Mimics Fault (f) Bit of Statusword (18)

The value of the selected Digital Output mimics bit 3 of the Statusword (f). When the “f” bit is high, the selected digital output will be high.

Mimics Switched On (so) Bit of Statusword (19)

The value of the selected Digital Output mimics bit 1 of the Statusword (so). When the “so” bit is high, the selected digital output will be high.

Mimics Ready to Switch On (rtso) Bit of Statusword (20)

The value of the selected Digital Output mimics bit 0 of the Statusword (rtso). When the “rtso” bit is high, the selected digital output will be high.

Mimics Operation Enabled (oe) Bit of Statusword (21)

The value of the selected Digital Output mimics bit 2 of the Statusword (oe). When the “oe” bit is high, the selected digital output will be high.

Mimics Voltage Enabled (ve) Bit of Statusword (22)

The value of the selected Digital Output mimics bit 4 of the Statusword (ve). When the “ve” bit is high, the selected digital output will be high.

Mimics Switch On Disabled (sod) Bit of Statusword (23)

The value of the selected Digital Output mimics bit 6 of the Statusword (sod). When the “sod” bit is high, the selected digital output will be high.

Mimics Quick Stop (qs) Bit of Statusword (24)

The value of the selected Digital Output mimics bit 5 of the Statusword (qs). When the “qs” bit is high, the selected digital output will be high.

Mimics Warning (w) Bit of Statusword (25)

The value of the selected Digital Output mimics bit 7 of the Statusword (w). When the “w” bit is high, the selected digital output will be high.

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

Parameters

Object ID	Name	Description
0x2211	GPIO output events	Timestamp at which the GPIO should trigger for “Timestamped Triggered Output”
0x2212	GPIO input actions	Time of the last rising edge for “Timestamped Rising Edge”
0x2213	GPIO position trigger	Sets Pulse Width, Position Setpoint and the encoder for “Position Triggered Output”

> Note:
>
> The pulse is generated independently of the direction from which the setpoint is crossed.

Object ID	Name	Description
0x60FE	Digital Outputs	Manufacture specific bits set the state of digital output
0x60FD	Digital Inputs	Manufacture specific bits show the state of digital inputs

Encoder Configuration

Parameter description

Function

Each encoder can have one of these functions: Commutation, Velocity or position.

It is of vital importance to have **exactly one** encoder configured for Commutation so that the motor shaft’s position is known to the commutation algorithm. In case your setup requires it, you can use a different encoder for Position or Velocity. In this case, the demanded and actual velocities will be scaled to the same reference frame of the position.

Resolution

For correct function it is necessary to provide the resolution of the encoder, this value is given in **Inc/Revolution**, please refer to your encoder's data sheet.

> Important:
>
> Please keep in mind that manufacturers sometimes give this specification in differing units.
>
> If the resolution according to your data sheet is not given in Inc/Revolution, please convert the value.

Zero velocity threshold

Time without a position change in which the velocity is assumed to be zero. After this time, the velocity value is set to 0 if no new position was received from the encoder. The default value is 5000 µs.

> Attention:
>
> Expert setting. The default value will perform sufficiently in most cases.

Polarity

The polarity represents the direction of rotation of the motor in comparison to the rotation of the encoder. If these directions don't match, then the polarity is inverted.

```
Set the value to -1 to invert the polarity.
```

Multiturn resolution

In case your encoder supports multiturn functionality, the number of stored revolutions must be specified here.

Parameters

Object ID	Name	Description
0x2110	Encoder 1 configuration	Configuration object for Encoder 1

Object ID	Name	Description
0x2111	Encoder 1 feedback	Provides the position or velocity value read at Encoder 1
0x2112	Encoder 2 configuration	Configuration object for Encoder 2
0x2113	Encoder 2 feedback	Provides the position or velocity value read at Encoder 2
0x2012:9	Position controller: Encoder source	Which encoder should be used for position control
0x2011:5	Velocity controller: Encoder source	Which encoder should be used for velocity control
0x2010:12	Torque controller: Encoder source	Which encoder should be used for torque control and for commutation

Encoder sampling frequency

Overview

The sampling frequency of BiSS, SSI or Nikon A-format encoders can be selected in the Objects 0x2110: Encoder 1 configuration and 0x2112: Encoder 2 configuration, subindex 28 Sampling frequency.

The sampling frequency can be set to a value between 16 or 32 kHz.

> Note:
> This is only available since firmware 5.0.7 In older versions, the sampling frequency is 16 kHz.

Limitations

If the encoder can't be sampled at the desired frequency, the error EncSmplT will be raised.

Some encoders only allow a particular frequency:

- Circulo internal encoders with multiturn support only 16 kHz.

In all other cases, the Circulo internal encoders will always be sampled at 32 kHz.

Parameters

Object ID	Name	Description
0x2110	Encoder 1 configuration	Subindex 28 selects the sampling frequency for encoder 1
0x2112	Encoder 2 configuration	Subindex 28 selects the sampling frequency for encoder 2

Commutation with Hall and ABI combined

Overview

An ABI encoder and a Hall sensor can be combined to function as a single encoder for commutation. This combines the benefits of a high resolution angle from the ABI encoder, with the absolute electrical position provided by the Hall sensor.

Requirements

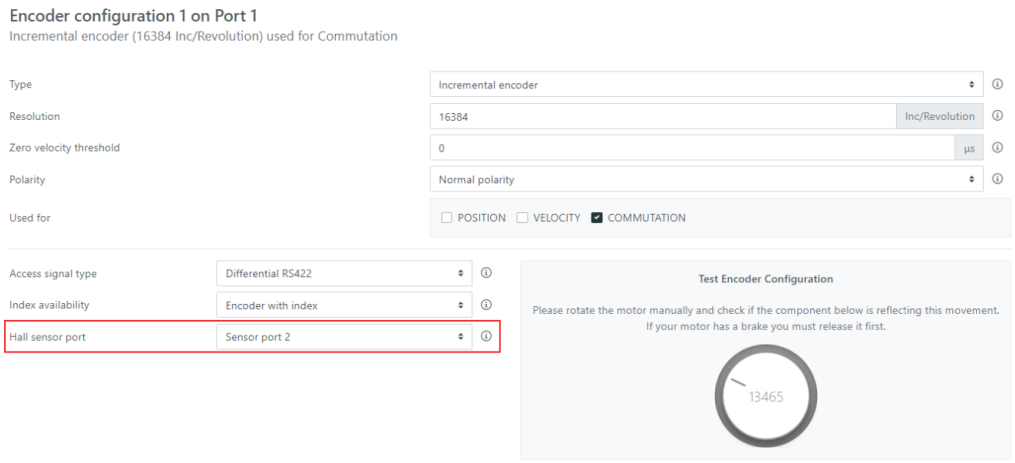
- The Drive needs a Hall sensor and an ABI encoder connected at two different connectors, both must be properly configured.
- The motor phases and the Hall pins have to be connected according to the Servo drive's documentation.
- The ABI commutation offset has to be found before this feature starts working.

Usage

To configure the HALL+ABI feature, the incremental encoder needs to be configured for at least "commutation" then specify the encoder port that the HALL sensor is connected to in the object "Encoder1 configuration Hall sensor port "0x2110:23"

The drive will automatically use it on startup to avoid the index detection routine. There is no need for additional configuration of the hall sensor on encoder port 2.

Configuring the Hall additionally on port 2 will lead to a HwRsrcEr.



Detailed description

Startup

When the feature is started, the Hall sensor is sampled and it is checked if it's in a valid state. If an invalid hall state [0, 0, 0] or [1, 1, 1] is found, an error 'HallSeq' is raised, and the connections should be checked.

Error Detection

Cable errors can lead to problems: if only one Hall sensor is disconnected or damaged, or if the phase signals are reversed, the problem will not be detected. Running the motor with Hall set as the only Commutation sensor can be done to ensure the integrity and proper connection of the Hall signals before integrating the ABI encoder.

Runtime

If no error is raised, the initial electrical angle is estimated based on the Hall sensors, and the Hall sensors are not sampled again. As soon as the incremental encoder passes the index, the service realigns the electrical angle to the formerly saved precise value.

Input counter

Overview

This feature allows counting pulses in a particular pin of an encoder connector (input counter pin).

It counts the transitions on the input counter pin at a maximum frequency of 16 kHz. The value of the counter is written to object 0x2040.

The input counter will be enabled by default in case the configuration allows it (for limitations see below).

Either a single-ended TTL line or a differential RS422 line can be used for the input counter.

Drive	Conditions for using a Single-ended (TTL) input counter	Conditions for using a differential (RS422) input counter
SOMANET Node	Encoder connector 2 doesn't have any encoder configured. Pulses are counted at pin 7.	Encoder connector 2 has an SSI encoder configured. Pulses are counted at pin 7 (-) and 8 (+).

Use case: Counting revolutions for simple speed estimation

In applications that run only in one direction (e.g.: a fan), it may be sufficient to count the pulses for calculating the speed.

Limitations

- At the moment, this feature can only be used if less than 2 encoders are configured or if one of the encoders is an SSI.
- If only one SSI encoder is connected (the other port being empty), the feature will use the empty port.
- The feature can be used at a maximum frequency of 16 kHz.

Parameters

Object ID	Name	Description
0x2040	Input counter	The value of the counter will be written to this object

Addressable LED control

Overview

This feature allows controlling a strip of addressable LEDs through the Digital I/O port to address signals in robots/actuators without additional Hardware.

It can update up to 100 LEDs at 13.28 Hz, all of them with the same value. The timings selected for the addressable LED protocol are:

- T0H = 310 ns
- T1H = 620 ns

- T0L = 940 ns
- T1L = 630 ns

A sequence of 24-bit is sent through the Digital I/O to control the brightness and color. The data transfer protocol uses a single NRZ communication mode.

The composition of 24-bit Data is as follows:

Procedure

- The feature can be configure in the GPIO editor menu in OBLAC Drives
- Or the values can be set directly using object 0x2215

One Digital I/O pin can be used for the addressable LED feature. The available voltage at the Digital I/O connector supplies with 250 mA either 3,3 V or 5 V.

> Note:
>
> The current from the 5 V output is shared with other ports (e.g. the external encoders) and is limited to 250 mA.

The addressable LED feature can be selected in OBLAC Drives in the GPIO editor menu, or the values can be entered in the object 0x2215 directly. The pin that shall be used must then be selected in object 0x2214.

Brightness and colors

The Digital I/O can deliver 250 mA, but the colors of the LEDs are limited in brightness at this current and could start to blink if more current is required.

Otherwise, an external power supply can be added, to achieve full brightness of the LEDs.

To control the color and brightness

1. go to the search parameters bar
2. type the object address 0x2215:01
3. Set the values (the values for this object can be typed in decimal or hexadecimal, as shown in table x.)

The table shows the parameter values to achieve the rated supported current of Circulo (250 mA) with a ring with 42 LEDs WS2812B-Mini-V3.

	Object 0x2215:01	
Condition	Value Dec	Value Hex
Blue	128	0x80
Red	32768	0x8000
Green	8388608	0x800000
White	3487029	0x353535

To achieve other colors, the values of the green, red and blue channels must be mixed as in the following scheme:

Brightness is controlled with a Hexadecimal value from 01 to FF

Green channel: 0xFF0000

Red channel: 0x00FF00

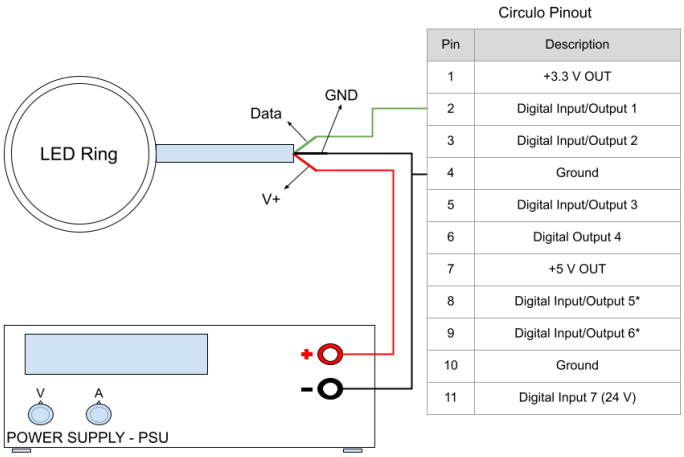
Blue channel: 0x0000FF

> Attention:
> When the current from the Digital I/O is used, the LED may be limited in brightness. To achieve full brightness, an external power supply must be attached.

Full brightness - current from an external power supply

For full brightness, an external power supply needs to be connected, as shown in image 1.

GND of the Digital I/O (Pin 4 or Pin 10) must be connected to the GND of the external power supply, as illustrated.



Circulo Pinout

Pin	Description
1	+3.3 V OUT
2	Digital Input/Output 1
3	Digital Input/Output 2
4	Ground
5	Digital Input/Output 3
6	Digital Output 4
7	+5 V OUT
8	Digital Input/Output 5*
9	Digital Input/Output 6*
10	Ground
11	Digital Input 7 (24 V)

Image 1 - LED ring powered up with an external power supply

Limitations

The value of the Digital Output can be set globally to 3.3 V or 5 V.

Ensure you set it to a voltage that is supported by your LED, or use an external power supply.

Parameters

Object ID	Name	Description
0x2210	GPIO pin con-figuration	Select which pin to use for the LED feature
0x2214:2	GPIO global option-s:Voltage level	Select the global voltage level for all DIO pins
0x2215	GPIO addressable LED control	Subitem 1: LED color selects the color in which the LED shall shine

Protection

Overview

This feature allows users to define limits for the following values:

- Undervoltage
- Overvoltage
- Overcurrent

Once these limits are violated, an error is triggered and the drive is stopped.

Also, a limit for Overtemperature triggers the error but this value is not customizable.

The temperature of the core board and the inverter are also measured and the values are available through objects 0x2030 (Core temperature) and 0x2031 (Drive temperature). The unit is given in m°C.

When the measured temperature approaches 95% of the permissible value, a warning is triggered in the Error Report Object. When the measured temperature reaches the limit, an error is triggered and the drive is stopped.

Usage

To set the value for user-defined protection limits, enter the value in the corresponding subitems of object 0x2006 Protection

- Undervoltage: 0x2006:1, Unit given in V
- Overvoltage: 0x2006:2, Unit given in V
- Overcurrent: 0x2006:3, Unit given in mA

Parameters

Object ID	Name	Description
0x2006	Protection	Limits for Undervoltage, Overvoltage and Overcurrent
0x2030	Core temperature	The temperature measured on the Core board
0x2031	Drive temperature	The temperature measured near the inverter

Errors and Warnings

To facilitate error diagnostics, SOMANET features several mechanisms:

Error codes thrown by the firmware: SOMANET throws firmware errors as specified in CiA 402.

Error Report Object: Each occurring error is described in the Error Report Object.

Remedies for protection errors: How to resolve issues with over-current, over-voltage, under-voltage and over-temperature.

Error codes thrown by the firmware

Object 0x603F throws error codes as standardized in CiA 402:

Object ID	Name
0x0000	NO_ERROR
0x2220	DEVICE_INTERNAL_CONTINUOUS_OVER_CURRENT
0x2250	DEVICE_INTERNAL_SHORT_CIRCUIT
0x2351	I2T_THERMAL_STATE_LOAD_LEVEL_WARNING
0x3130	PHASE_FAILURE
0x3131	PHASE_FAILURE_L1
0x3132	PHASE_FAILURE_L2
0x3133	PHASE_FAILURE_L3
0x3210	DC_LINK_OVER_VOLTAGE
0x3220	DC_LINK_UNDER_VOLTAGE
0x3331	FIELD_CIRCUIT_INTERRUPTED
0x4210	EXCESS_TEMPERATURE_DEVICE
0x4310	EXCESS_TEMPERATURE_DRIVE
0x5300	OPERATING_UNIT
0x6010	SOFTWARE_RESET_WATCHDOG
0x6320	PARAMETER_ERROR
0x7121	MOTOR_BLOCKED
0x7300	SENSOR

Object ID	Name
0x7500	COMMUNICATION

Note:

Whenever a firmware error occurs, there is a dedicated entry in the Error Report Object which will specify what triggered the error.

Object ID	Name	Description
0x603F	Error Code	Provides the error code of the last error which occurred in the drive device

Error Report Object

Provides further information about the error or warning reported by the Error code object 0x603F.

No fault

Error code	Description	Explanation	Error code from object 0x603F	Remedy
NoFault	No fault	Default value of the object, which means no fault is to be reported.		This error code should indicate that there's no error reported, thus the drive is not in fault state. If the drive is shown in fault state and this error code is reported, possibly

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				this is caused by a fault in master/slave communication for example in the VM version OBLAC Drives. Rebooting the VM would solve the problem in this case. Else please contact technical support of Synapticon.

Protection faults of user-defined parameters

Errors triggered by user-defined thresholds (for more details see object 0x2006: Protection)

Error code	Description	Explanation	Error code from object 0x603F	Remedy
PuOcA	Protection User-defined Over-current Phase A	The current sensor at drive phase A/B/C sensed a current exceeding the	0x2220	As the threshold is user-defined, please first check the object 0x2006:3 Overcurrent setpoint to make sure that the threshold is set

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		user-defined Over-current threshold (0x2006:3 Protection Over-current setpoint).If the user-defined threshold is higher than the hardware protection threshold, as long as the hardware protection is available, this software protection can be considered unfunctional.		appropriately.Regarding an over-current, please refer to the remedy for over-current.
PuOcB	Protection User-defined Over-current Phase B		0x2220	
PuOcC	Protection User-defined Over-current Phase C		0x2220	
PuOt	Protection User-defined Over-temperature		0x4210, 0x4310	
PuUv	Protection User-defined	Indicates that	0x3220	As the threshold is

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	Under-voltage	the DC bus voltage is measured below the user-defined threshold (0x2006:1 Protection Under-voltage setpoint) when the drive is in a CiA 402 state that requires high-level power to be applied. These states are Switched on, Operation enabled, Quick stop active and Fault reaction active).		user-defined, please first check the object 0x2006:1 Under-voltage setpoint.Then please refer to the remedy for under-voltage.
PuOv	Protection User-defined	Indicates that the DC bus	0x3210	As the threshold is user-defined, please

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	Over-voltage	voltage is measured exceeding the user-defined threshold (0x2006:2 Protection Over-voltage setpoint).		first check the object 0x2006:2 Overvoltage setpoint. Then please refer to the remedy for over-voltage.

Hardware Protection Faults

Errors triggered by hardware built-in protection logic.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
PhUdef	Protection Hardware Undefined	Indicates a malfunction of the protection circuit.	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				investigation.
PhFault	Protection Hardware Fault	A generic hardware fault was detected. In Circulo drives this can be caused by any of the following: phase over-current, under-voltage, over-voltage, over-temperature, deadtime error, shoot-through or watchdog timer.	0x5300	Please refer to the Protection hardware remedies for over-current, under-voltage, over-voltage, over-temperature, deadtime phase and watchdog.
PhWtdg	Protection Hardware Watchdog	The watchdog is a service, by checking periodically with a timer, guaranteeing that the drive control logic is working in the proper fre-	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		quency. This error indicates that the watch-dog tick is not received on time by the hard-ware protection, thus the drive is not working properly.		investigation.
PhDeadTA	Protection Hardware Deadtime Phase A	Deadtime is a common pro-tection strategy in PWM control for avoiding bridge shoot-through, a short-circuit accident caused by tran-sistors on both sides being open at the same time. This error indicates that the effect-ive deadtime is	0x2250	Please note down the information regarding the occurence of the error and reach Syn-apticon tech-nical support for further investigation.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		detected below the man-ufacturer-defined threshold.		
PhDeadTB	Protection Hardware Deadtime Phase B		0x2250	
PhDeadTC	Protection Hardware Deadtime Phase C		0x2250	
PhDeadTD	Protection Hardware Deadtime Phase D		0x2250	
PhOvUvOt	Protection Hardware Over-voltage / Under-voltage / Over-tem-perature	The drive tem-perature which measured close to power MOSFETs and the DC bus voltage are checked by the same hardware protection cir-cuit against manufacturer-defined	0x3331	Please refer to the remedy for over-voltage, under-voltage and over-tem-perature.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		thresholds. This error indicates that one or more of the three errors is/are triggered.		
PhOc	Protection Hardware Over-current	Indicate that the phase current or the DC bus current is measured exceeding the manufacturer-defined threshold (referring to the hardware document, link), which triggers the hardware protection circuit.	0x2220	Please refer to the remedy for over-current.
PhBrkOc	Protection Hardware Brake Over-current		0x5300	Please refer to the remedy for brake over-current.
PhGndOc	Protection		0x5300	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	Hardware GND Over-current			
PhUv	Protection Hardware Under-voltage		0x3220	Please refer to the remedy for under-voltage.
PhOv	Protection Hardware Over-voltage		0x3210	Please refer to the remedyfor over-voltage.
PhOcA	Protection Hardware Over-current Phase A		0x2220	Please refer to the remedy for over-current .
PhOcB	Protection Hardware Over-current Phase B		0x2220	Please refer to the remedy for over-current.
PhOcC	Protection Hardware Over-current Phase C		0x2220	Please refer to the remedy for over-current.

Safety Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SfeDiIvd	Safety Digital Input Invalid	Safety module reports a discrepancy between the two STO-SBC input channels. The error will be shown with a 100 ms interval.	0x5300	Check and fix the safety inputs wiring accordingly with the assistance of tools for example with multimeter measurement. Compare the measured inputs with 0x2611 (Safety digital input diagnostics) and locate the signal leading to the fault. Note that You can only reset the safety fault if you are in the SAFETY ACTIVATED state, meaning both STO-SBC1 (0x2611:1) and STO-SBC2

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				(0x2611:2) inputs shall be in the Low state when you are trying to clear the fault. Please refer to the documentation about the safety functions and the STO/SBC inputs wiring of your hardware and check the use of the safety functions in our documentation.
SfeFault	Safety Fault	A safety-based hardware fault detected.	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further investigation. If this fault is permanent and can't be reset this

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				is a fault caused by broken internal hardware

BiSS Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BisErBit	BiSS Error Bit active.	One of the bits in the BiSS frame is used to signal error. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the error bit being active.	0x7303, 0x7304	Please check the 2 cases often seen:1. Mis-configuration of Status bits active value. The status bits (error bit and warning bit) are by default read as active low, i.e. the error bit being 0 indicates the error being active. Please check the documentation of

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BisWnBit	BiSS Warning Bit active.	One of the bits in the BiSS frame is used to signal warning. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the warning bit being active.	0x7303, 0x7304	the encoder and correctly configure the reading.2. The encoder is indeed rasing an error/ warning. Please check the documentation of the encoder, consult the encoder manufacturer for further investigation.
BisAcBit	BiSS Acknowledge Bit not received.	The BiSS slave is not acknowledging the request to communicate sent by the drive. This usually happens when there is some	0x7303, 0x7304	Please firstly make sure that the encoder is connected to the port which is being configured. Then check the wiring accoding to the

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		issue with the power or communication link, e.g. one of the wires is broken or not well connected.		hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
BisSrtBt	BiSS Start Bit not received.	The BiSS slave acknowledged the request to communicate, but it didn't sent the start bit to start the communication. This usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	0x7303, 0x7304	
BisSloLo	BiSS data (SLO) is per-	The SLO line	0x7303, 0x7304	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	manently Low.	should be set to high by the BiSS slave when it is ready to establish communication. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.		
BisFrame	BiSS Frame configuration error.	The sensor configuration causes an issue related to the BiSS frame. This error is currently raised only if the BiSS frame size is higher than 64 bits.	0x7303, 0x7304	Please check the sensor configuration according to the documentation of the encoder manufacturer. The most relevant objects to check:1. Resolution, unit Inc/

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				Revolution2. Multiturn resolution, unit number of bits3. Number of filling bits, unit number of bits
BisCrc	BiSS CRC error.	The BiSS frame contains a CRC6 of some of the data that is being sent. This error is raised when the CRC sent by the BiSS slave is different than the CRC computed by the drive. This usually means that the drive can't read the frame properly.	0x7303, 0x7304	Regarding a CRC error, there are several causes often seen:1. Misconfiguration of the sensor reading leading to the bits being read wrong. In this case the fault cannot be reset. Please check the sensor configuration according to the documentation of the encoder

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				manufacturer. Common parameters wrongly configured are the singleturn resolution and multiturn resolution.2. Misconfiguration of the object CRC polynom, thus the CRC is calculated wrong. In this case the fault cannot be reset. Please check the polynom in the encoder documentation. The value by default: 48 represents x^0 + x^1 + x^6, which usually doesn't need to be modified.3. Inconsistency in

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
BisRegEr	BiSS register communication error	Some BiSS encoders require the firmware to read/write some of their registers to be initialized	0x7303, 0x7304	Try to reconfigure the encoder that raised this error. To do so, there are two possibilities:- Set its

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		properly. This error is triggered when any of these BiSS register transaction fail.		function to Disabled and then back to the function it had.- Remove the encoder from the sensor port list (0x2100) and add it again.

SSI Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SsiFrame	SSI Frame configuration error.	The sensor configuration causes an issue related to the SSI frame. This error is currently raised only if the SSI frame size is higher than 256 bits.	0x7303, 0x7304	Please check the sensor configuration, especially the object Frame size according to the documentation of the encoder manufacturer.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SsiDtaLo	SSI Data is permanently Low.	The data line should be set to high by the SSI slave when it is ready to establish communication. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	0x7303, 0x7304	Please firstly make sure that the encoder is connected to the port which is being configured. Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
SsiDtaHi	SSI Data is permanently High.	The data line should be set to low by the SSI	0x7303, 0x7304	Please firstly make sure that the encoder is

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		slave during the monoflop time, which happens right after clocking the complete SSI frame. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.		connected to the port which is being configured. Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
SsiChksm	SSI Checksum error.	Some SSI slaves include a ckecksum of the data in the SSI frame. Currently our drives can	0x7303, 0x7304	Regarding a checksum error, there are several causes often

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		be configured to read a parity bit from compatible SSI encoders. This error means the parity bit computed by the firmware is different than the one sent by the SSI slave.		seen:1. Mis-configuration of the sensor lead-ing to the bits being read wrong or the checksum being calculated wrong. In this case the fault cannot be reset. Please check the sensor con-figuration accord-ing to the documentation of the encoder manufacturer.2. Inconsistency in signal transfer. In this case the fault can some-times be reset, but would occur under certain cir-cumstances. Please check the

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				eration for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.

QEI Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
QeiLsTck	QEI Lost Ticks.	The number of ticks counted in two successive index pulses is different from the sensor resolution configured by user.	0x7303, 0x7304	There are several causes often seen: 1. The configured resolution is wrong. Please check the Resolution configured according to the encoder datasheet, and please pay attention to the conversion between units (Resolution = Cycles per Revolution * 4). 2. There's a loss of pulses in encoder reading, which can be due to: 2a. The encoder

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				being improperly mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and consult the encoder manufacturer for advices. 2b. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influ-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				oldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transferred with acceptable noise. 2c. The pulses are read with an exceeding frequency. This case rarely happens, but can be possible if the encoder resolution and/or the spin speed being unusually high.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
QeiNoIdx	QEI No Index.	For an incremental encoder configured to be with 3 channels (usually known as ABI/ ABZ), the drive is expecting to receive an Index pulse which provides an index for the absolute singleturn position. This error indicates that the Index pulse couldn't be detected, while the drive moved the motor in order to detect it.	0x7303, 0x7304	Please first check whether your encoder is indeed with the Index pulse, i.e. is an ABI/ ABZ encoder. If it is an encoder without index, please choose 2-channel (no index) for Number of channels. Else, most probable causes are: 1. The encoder being improperly mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				consult the encoder manufacturer for advices. 2. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transferred with acceptable noise.
QeiOpnWr	QEI Open Wire.	Indicate that one of the differential A or B lines is disconnected. Only available on hardware that supports it.	0x7303, 0x7304	Please check the wiring with for example the diode mode of a multimeter, making sure that the wires are stably transferring the signals.

Hall Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
HallSeq	Hall Sequence error.	The hall sensor outputs	0x7303, 0x7304	Please check

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		have an invalid value (111 or 000), or the sequence between the current state and the last state is invalid.		the wiring and make sure that the signals are transferred with good consistency. If the HALL sensor is used together with an incremental encoder for commutation, it can be that the HALL sensor A/B/C pins are connected in a wrong order. Please verify the cabling, and verify the order by running offset detection with only the HALL sensor configured for commutation. If the order is correct, the result of the offset detection

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				should be around 0, and 0x2003:5 Motor phases inverted should show 0 (No).

A-Format Sensor Fault

Error code	Description	Explanation	Error code from object 0x603F	Remedy
AftFrame	A-Format Frame error.	The fixed data in the recieved A-Format frame (like sync code) is wrong. This usually means that the A-Format transmission speed is wrong.	0x7303, 0x7304	Please check the transmission speed configuration according to the encoder datasheet.
AftTmout	A-Format Timeout.	The A-Format encoder didn't start communication after sending a	0x7303, 0x7304	Please check the wiring and make sure that the signals are transferred with

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		request.		good consistency. Use a oscilloscope to monitor the signal transferred for further investigation.
AftCrc	A-Format CRC error.	The A-Format frame contains a CRC of some of the data that is being sent. This error is raised when the CRC sent by the encoder is different than the CRC computed by the drive. This usually means that the drive can't read the frame properly, which usually happens because of noise in the com-	0x7303, 0x7304	Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situ-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		munication link.		ation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transferred with acceptable noise.
AftBatt	A-Format BATT status.	The encoder is raising Battery Alarm. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Battery Alarm. The usual case is that the battery voltage is low, the battery is wrongly wired, or there's no battery connected.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
AftMTErr	A-Format MTERR status.	The encoder is raising MT Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the MT Error.
AftOvFlw	A-Format OvFlow status.	The encoder is raising Over Flow Alarm. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Over Flow Alarm.
AftOvSpd	A-Format OVSPD status.	The encoder is raising Over Speed Alarm. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Over Speed Alarm. The usual case is that the rotation speed exceeds the speed limit

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				of the encoder.
AftMemEr	A-Format MEMERR status.	The encoder is raising Memory Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Memory Error.
AftSTErr	A-Format STERR status.	The encoder is raising ST Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the ST Error.
AftPSErr	A-Format PSERR status.	The encoder is raising PS Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the PS Error.
AftBusyF	A-Format BUSY status.	The encoder is raising BUSY Flag. Check your	0x7303, 0x7304	Please check the encoder documentation for

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		A-Format encoder data-sheet for more information.		better under-standing the meaning of the BUSY Flag.
AftMemBs	A-Format MemBusy status.	The encoder is raising MEMBUSY Flag. Check your A-Format encoder data-sheet for more information.	0x7303, 0x7304	Please check the encoder doc-umentation for better under-standing the meaning of the MEMBUSY Flag.
AftOvTmp	A-Format OvTemp status.	The encoder is raising Tem-perature Warn-ing. Check your A-Format encoder data-sheet for more information.	0x7303, 0x7304	Please check the encoder doc-umentation for better under-standing the meaning of the Temperature Warning. In case of over-temperature please refer to the remedys for over-tem-perature.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
AftIncEr	A-Format IncErr status.	The encoder is raising Incremental Signal Error. Check your A-Format encoder datasheet for more information.	0x7303, 0x7304	Please check the encoder documentation for better understanding the meaning of the Incremental Signal Error.

Generic Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SnsrAngl	Sensor Angle error.	1. The electrical angle was not completing a whole revolution during offset detection. 2. The calculated commutation angle is out of bounds. A sing-	0x7300, 0x7303, 0x7304	Please firstly check whether the motor rotates during offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. Please check

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		leturn related configuration parameter (resolution, singleturn) with a wrong value normally causes this error. 3. The motor is not moving during the QEI index detection.		the encoder configuration, and make sure that while rotating the motor for one full turn, the encoder value is correctly responding to the mechanical position. You can manually rotate the motor for investigation. An alternative way is with OBLAC opening another page and ploting the position feedback of the encoder used for commutation while running offset detection.
SnsrCfg	Sensor Configuration error.	1. Some	0x6320, 0x7303,	Please check the configuration

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		encoder parameter was set with an invalid value in the Object dictionary. 2. An action that required a particular encoder function (position, velocity or commutation) was triggered, but there is no encoder configured for that function. 3. There was an issue when trying to initialize or configure an encoder.	0x7304	of the sensors, especially the parameters changed lately. Make sure that all functions required (position, velocity, commutation) are configured with the sensors.
HwRsrcEr	Hardware Resource Error.	The software configuration is invalid for the	0x7303, 0x7304	Please check the sensor configurations and

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		hardware. For example, a sensor configured is with a sensor type not supported by that port.		the digital IO settings, making sure that the setting is supported by your hardware according to the documentation.
IvldGpio	Invalid Gpio configuration.	The GPIO related configuration chosen is not available. This includes voltage level, direction of the GPIO, or GPIO trigger.	0x7303, 0x7304	Please check the configuration of the GPIO according to the documentation.
SnsrBatt	Sensor battery error	This error is raised when an encoder battery was expected, but couldn't be detected. This error is raised	0x7303, 0x7304	Please check the documentation to make sure the battery is connected properly and power cycle. In case there is

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		only by multiturn encoders that need a battery, like Circulo's Internal encoder 1 when it's configured as a multiturn encoder. The error is also raised when the multiturn count of such an encoder configured for multiturn is lost.		no battery or if the battery is not needed, select "0" or "No" to the "Multiturn resolution" parameter of this encoder.
SkpCycls	Serial encoder service is skipping cycles	The error SkpCycls is raised when a serial encoder service (REM16, BiSS, SSI, A-Format, MA302) is looping slower than	0x7303, 0x7304	Minimize the time that is required for reading the encoder. In a BiSS encoder the following approach can be used:

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		16 kHz, which means that the cycles are taking longer than 62.5 us. The warning SkpCycls is raised when an encoder that supports 32 kHz sampling (which are only BiSS or SSI) is looping at 16 kHz instead.		First the clock frequency should be increased, and then the encoder timeout should be reduced (determine the minimum value from the encoder datasheet). For an SSI encoder, only the clock frequency can be increased because the timeout is fixed. For other encoders, the speed is fixed. Please contact support if this issue occurs.
EncMtLos	Encoder multiturn was lost	This error is raised when the multiturn	0x7303, 0x7304	Please check the documentation to

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		count of a multiturn encoder that needs a battery is lost, like Circulo's Internal encoder 1 when it's configured as a multiturn encoder. This situation will happen when the battery of the encoder is low or removed and the drive is power-cycled.		make sure the battery is connected properly and power cycle. In case there is no battery or if the battery is not needed, select "0" or "No" to the "Multiturn resolution" parameter of this encoder.

Brake Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BrkNtRls	Brake release failed	Brake release	0x7121	Please firstly

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		procedure for pin brake failed, i.e. the configured displacement threshold (0x2004:8 Minimum displacement (pin brake)) is not reached with the configrued torque (0x2004:9 Maximum torque (pin brake)).		check whether the motor functions correct, for example by manually releasing the brake and running offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. While the motor functions correct, please further tune the two configuration parameters, by increasing the torque or decreasing the minimum displacement, until the brake can be released stably

Error code	Description	Explanation	Error code from object 0x603F	Remedy
BkHiPull	Pull brake voltage too high	Pull brake voltage (0x2004:1 Pull voltage) higher than measured DC bus voltage (0x6079: DC link circuit voltage).	0x6320	Please measure the main power supply voltage supplied to the drive, and check whether the result is close to 0x6079: DC link circuit voltage. Pay attention that the actual voltage supplied to the drive would be slightly lower than the voltage output by the power supply, because of attenuation. Check the configuration of the pull/hold voltage, which should be lower than the DC Bus voltage.
BkHiHold	Hold brake voltage too high	Hold brake voltage (0x2004:2 Hold voltage) higher than measured DC bus voltage (0x6079: DC link circuit voltage).	0x6320	

Parameter Fault

Error code	Description	Explanation	Error code from object 0x603F	Remedy
IvldOfst	Invalid offset	The commutation offset in 0x2001: Commutation angle offset is not valid. This error comes directly from reading object 0x2009:1 Commutation offset: State.	0x6320	Please run offset detection before operations. Please pay attention that, the change in some fundamental configuration parameters (sensor configuration, motor configuration) can make the existing offset detection result invalid. Please redo offset detection in that case.
IvldOpmd	Invalid operational mode	Requested opmode is not supported. Set a valid value in 0x6060: Modes of operation.	0x6320	Please check our documentation for the available operation modes. Please check 0x6060: Modes of operation, and

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				make sure that the master controller is sending a correct command.
ZeroMxI	Max. current is zero or negative	User-defined limit on motor max current (0x6073: Max current) is 0.	0x6320	Please configure the 0x6073: Max current to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxT	Max. torque is zero or negative	User-defined limit on motor max current (0x6072: Max torque) is 0.	0x6320	Please configure the 0x6072: Max torque to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxMS	Max. motor speed is zero or negative	The value of 0x6080: Max motor speed is 0, so no motion is allowed.	0x6320	Pleaes configure 0x6080: Max motor speed to a valid value larger than 0, according

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				to your motor datasheet or your application need.
ZeroMxP	Max. power is zero or negative	User-defined limit on Max. power (0x200B) is 0.	0x6320	Pleaes configure 0x200B: Max power to a valid value larger than 0, according to your motor datasheet or your application need.
IvldPara	Invalid Parameter	The value of a parameter is invalid.		Please check the values changed lately according to the documentation.
SwLimOut	Software Limit Out.	0x607D: Software position limit is outside of encoder range limit.	0x6320	Please check object 0x607D: Software position limit and make sure that it is inside of 0x607B: Position range limit.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
SiVelPfx	Invalid velocity scaling factor	The velocity scaling prefix defined by 0x60A9: SI unit velocity is outside of the supported range.	0x6320	Please check OBLAC Drives, which directly provides available options for the configuration of 0x60A9: SI unit velocity, and the documentation for user defined velocity units. Make sure that a valid data is write to the object.
SiVelTim	Invalid SI velocity time factor	The velocity scaling prefix defined by 0x60A9: SI unit velocity is outside of the supported range.	0x6320	Please check OBLAC Drives, which directly provides available options for the configuration of 0x60A9: SI unit velocity, and the documentation for user defined velocity units. Make

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				sure that a valid data is write to the object.
IvldFeed	Invalid Feed constant	One of the Feed parameters 0x6092 (Feed or Shaft revolutions) is set to 0.	0x6320	Please check the values of 0x6092:1 and 0x6092:2.

Motion Control Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
MxPwrLmt	Maximum power limit	Maximum power limit is active. Torque set point is limited.	0x5200	
FollwErr	Following error		0x8611	
MxTorLmt	Maximum torque limit	Maximum torque limit is active. Target or demand torque is lim-	0x5200	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		ited.		
MxMSLmt	Maximum motor speed limit	Max. motor speed limit is active. Target or demand velocity is limited. Also, it is active when Profile velocity in Profile postiion mode is limited.	0x5200	
SwPosLmt	Software position limit	Software position limit is active. Target position is limited or Actual position is limited in order to prevent runaway of a motor.	0x5300, 0x8612	
PosRgLmt	Encoder range limit	Encoder range limit is active. Target position is limited and can't wrap around range	0x8612	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		limits.		
IntgrLmt	IntLmAct	Integral limit is active. Integral part of position or velocity PID controller is limited.	0x5200	
PosLmAct	Positive Limit Switch Active	Motor is at the position limit switch and can't move further.	0x5300	
NegLmAct	Negative Limit Switch Active	Motor is at the negative limit switch and can't move further.	0x5300	

Communication Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
CyclicHb	Cyclic heartbeat	A communication keep-	0x7500	If the inactivity of

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		alive heartbeat was dropped. Check that the PDO messages are being sent by the master cyclically, and within 100 ms of each other.		communication is known/ intended, the occurrence of the error is expected. Do [Fault reset] to clear the fault after the communication is reestablished. Otherwise, the occurrence of the error indicates that there exists unknown issues in the communication chain. Possible causes include: - communication delay in master program, check the running status of the master machine. - disturbance in communication signal transfer,

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				make sure that a communication cable with shielding is used or try with the cable provided by Synapticon, check the stability of cabling.

Internal Errors

Error code	Description	Explanation	Error code from object 0x603F	Remedy
Intern01	Internal fault 1	Internal Error Code. Please contact support.	0x5300	Please note down the information regarding the occurrence of the error and reach Synapticon technical support for further investigation.
Intern02	Internal fault 2	Internal Error Code. Please contact support.	0x5300	
Intern03	Internal fault 3	Internal Error Code. Please contact support.	0x5300	

Error code	Description	Explanation	Error code from object 0x603F	Remedy
Intern04	Internal fault 4	Internal Error Code. Drive service is skipping cycles and there is no downstream communication.	0x7500	Please contact support.
Intern05	Internal fault 5	Internal Error Code. Motion service is skipping cycles and there is no downstream communication.	0x7500	Please contact support.

Filter cutoff frequency bounds

Error code	Description	Explanation	Error code from object 0x603F	Remedy
VeFiFcLo	Velocity filter cut-off frequency too low	0x2021:2 Velocity feedback filter Cutoff frequency is lower than the minimum limit.	0x6320	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is con-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				figured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
VeFiFcHi	Velocity filter cut-off frequency too high	0x2021:2 Velocity feedback filter Cutoff frequency is higher than the maximum limit.	0x6320	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
PoFiFcLo	Position filter cut-off frequency too	0x2022:2 Position feedback filter Cutoff fre-	0x6320	Please check 0x2022:2 Pos-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	low	quency is lower than the minimum limit.		ition feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
PoFiFcHi	Position filter cut-off frequency too high	0x2022:2 Position feedback filter Cutoff frequency is higher than the maximum limit.	0x6320	Please check 0x2022:2 Position feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				the object for latest information.
VffFcLo	Velocity feed forward filter cut-off frequency too low	0x2015:2 Velocity feed forward Cutoff frequency is lower than the minimum limit.	0x6320	Please check 0x2015:2 Velocity feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
VffFcHi	Velocity feed forward filter cut-off frequency too high	0x2015:2 Velocity feed forward Cutoff frequency is higher than the maximum limit.	0x6320	Please check 0x2015:2 Velocity feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured lower

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				than the limit 2000 Hz. Please check the documentation of the object for latest information.
NhFiFcHi	Notch filter center frequency too high	0x2023:2 Notch filter Center frequency is higher than the maximum limit.	0x6320	Please check 0x2023 Notch filter, and make sure that the center frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
NhFiPara	Invalid Notch filter parameters	Invalid configuration was detected for the notch filter.	0x6320	Please check 0x2023 Notch filter, and make sure that the rejection band is set smaller than the center fre-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				quency.

General errors

Error code	Description	Explanation	Error code from object 0x603F	Remedy
InvConf	Invalid Configuration	The operation to save the config.csv file did not properly complete, and the config.csv may be corrupt. The config.csv file will not be loaded.	0x6320	This most likely happened as a result of power cycling the drive directly after issuing a command to save the configuration. A file named "lock" symbolizes that the save operation wasn't completed. If you reconfigure your drive and successfully save the configuration, oper-

Error code	Description	Explanation	Error code from object 0x603F	Remedy
				ation will be restored. You may also upload a known good config.csv file, delete the "lock" file, issue a "restore configuration" command.
InterLck	DI for Interlock (Fault Reaction) is low		0x5300	
SkpCycls	Motion or drive service is skipping cycles	The warning SkpCycls is raised when a motion or drive service is looping slower than 4 kHz, which means that the cycles are taking longer than 250 us.	0xF002	Try to disable some features (e.g. notch filter or position feedback filters) and see if the problem is fixed.
CtrlCyEx	Control cycle	This warning	0xF002	Try to disable

Error code	Description	Explanation	Error code from object 0x603F	Remedy
	time is exceeded	is raised when a Somanet control loop is executing slower than 4 kHz, which means that the cycles are taking longer than 250 us.		some features (e.g. notch filter or position feedback filters) and see if the problem is fixed.

Temperature Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
TmpCore or PhOtCore		This fault is triggered when the temperature of the core becomes too high and crosses the error threshold (> 100 °C). This happens usually	0x4210	Disable the drive until the temperature is reduced or use external cooling to bring down the overall temperature.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		because of sustained load on the processor, high operating frequency and even high ambient temperature. High core temperature can lead to equipment damage. It can also result in irregular clock timing.		
TmpDrive or PhOtDrve		This fault is triggered when the temperature of the drive becomes too high and crosses the error threshold (> 100 °C). This happens usually because of sustained load on the processor,	0x4310	Disable the drive until the temperature is reduced or use external cooling to bring down the overall temperature.

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		current leakage from the on-board com-ponenets, high operating fre-quency and even ambient temperature. High drive tem-perature can lead to equip-ment damage.		

External Sensor Faults

Error code	Description	Explanation	Error code from object 0x603F	Remedy
ExAnSnsr	External Scaled Ana-log Sensor Output exceeded one of the threshold val-ues.	This fault is triggered when the scaled external analog sensor output exceeds the upper level threshold or	0x7300	Please make sure that the upper and lower level threshold values are suit-able for the sensor you have connected and

Error code	Description	Explanation	Error code from object 0x603F	Remedy
		becomes lower than the lower level threshold. 0x2038:11 External scaled measurement Upper error threshold 0x2038:12 External scaled measurement Lower error threshold		that the values have been calculated according to the instructions given in the documentation for scaled measurements.

Diagnostic Errors

Error code	Description	Explanation	Error code from object 0x603F	Remedy
DiagErr	Diagnostics error	Error triggered by OS command 16	0x6320	

Remedies for protection errors

Remedy for over-current

There are various possible causes for an over-current error. Here are several potential ones which would be helpful to check:

- the controller output results in an exceeding current generated in the phase. Read the 0x6077 Torque actual value and the 0x6071 Target torque during operation, the scope of OBLAC can be used for this purpose. If an exceeding value can be seen in the torque value while the error happens, it often indicates that:
    - the PID controller (of the loop where the error is seen) is tuned too sharp to be stable. In this case, tune the controller softer following our tuning instructions. The controller can be easily unstable if the physical stability of the setup is bad, for example when the setup is not well fixed, or the load is with low rigidity.
    - the command sent to the drive is too sharp. Make sure that your command is sent with an appropriate acceleration / deceleration value. If a position step profile with relatively large amplitude or a quick drag of the slider in Playground-Position is given, this can be the case.
    - if the velocity value doesn't follow the torque value well (the torque doesn't lead to active acceleration) there may be a problem in the commutation angle calculation. Check the feedback value of the sensor and perform the offset detection again.
- an inappropriate power supply is used:
    - a voltage too high or too low would lead to malfunction of components. Check the voltage output of your power supply, if necessary with a measurement tool.
    - if the power output rating of the supply is not enough for the application, the voltage would drop while current is high, leading to the issue declared above.
- a short in the circuit can easily trigger over-current.
- a damaged hardware. In this case, there are often visible symptoms such as sparks or smoke.

Remedy for over-voltage

Possible causes for over-voltages include:

- the voltage output by the power supply is indeed exceeding the threshold. Adjust the voltage output accordingly.
- the voltage jump during motion leads to the error:

  When a load is causing the motor to turn in the opposite direction of the commanded motion, the motor generates power instead of consuming power. This leads to a voltage jump if there's no braking chopper in the circuit.

In this case use a power supply with integrated chopper or add a chopper device in the circuit for protection (e.g. Synapticon braking chopper).

Please also refer to our guidelines about regenerative energy.

Remedy for under-voltage

Possible causes for under-voltage include:

- the voltage supplied to the drive is continuously below the defined threshold.

> Note:
>
> It is normal to have a small (<1V) drop in voltage in the circuit due to resistance.

The power supply should be set to output a voltage slightly higher than the threshold or the threshold should be set lower in this regard. For better investigation, measure the voltage with a multimeter:

1. 0x6079 The DC link circuit voltage
2. The voltage between the main power supply pins on the drive
3. The voltage output by the power supply

The three results should be close. Otherwise, if there are large difference between the first two results, the hardware may be damaged. If the first two results are close, and the third is different, there may be a big voltage drop in the supply circuit due to factors such as cable resistance.

- If the error occurs during motion, the voltage output of the power supply drops when the maximum output current is reached. Check the specification of the power supply and make sure that a supply with adequate power rating is used.

Remedy for over-temperature

When the temperature of the drive is too high, it is usually a sign of inadequate cooling. First, please check the values in 0x2030 Core temperature and 0x2031 Drive temperature and note down the values. If the values seem much higher than a reasonable value, please contact Synapticon technical support. Please refer to our thermal installation guide for further information about improving the cooling.

Logging

Overview

Logging provides a list of events that happened within the firmware so that when an error or warning occurs, the user can trace back and see what might have triggered the error.

Operational activity logs of SOMANET systems can be generated and stored for bug-fixing, troubleshooting and diagnosis of systems.

The logs can be accessed via FoE:

- the current logs are saved to log_curr.log **log_curr.log** which has a capacity of 9 kB.
- When its memory is filled up, the logs are transferred to **log_prev.log** so that the latest 9 kB of logging history can always be retrieved.
- By concatenation of ‘log_curr.log’ and ‘log_prev.log’, between 9 kB and 18 kB of logging history can be retrieved.

Types

Logs are tagged with a “type” to facilitate quickly searching and finding important events within the log. These types are as follows:

DEBG	denotes a debugging log entry
INFO	this is the default log level. It contains information such as state changes
WARN	warnings, no fault is triggered
ERR	error, drive fault expected

All standard and SOMANET specific errors are listed in the section Errors and Warnings.

Format

Each line in the log has the format:

<time TYPE|><message>

If a message exceeds 105 characters, the message is truncated and a ~character is placed in the 105th position.

The time initially starts at *00:00:00.000* and is thus the time since last boot.

Motion Control

To make you motion system perform in an optimal way, it is necessary to understand the basic concepts behind motion via EtherCAT, more specifically the CiA 402 Drive Profile and the SOMANET specific extensions:

Control Loops

Current and Torque Control Loop

Velocity Control Loop

Position Control Loop

Dual Loop Cascaded Postion Control

Modes of operation

Cyclic modes

Cyclic synchronous torque mode (CST) In torque control mode the master sends desired torques to the drive.

Cyclic synchronous velocity mode (CSV) In velocity control modes the master sends desired velocities to the drive.

Cyclic synchronous position mode (CSP) In position control modes the master sends desired positions to the drive.

Profile modes

Profile position mode In profile position mode the trajectory to a target point is generated on the drive.

Profile velocity mode In profile velocity mode the motor will rotate at a target velocity.

Profile torque mode In profile torque mode the drive sends a target torque to the motor.

Homing modes

Diagnostics opmode Used for triggering diagnostic OS commands.

Selecting and switching the Modes Describes how the modes can be selected.

Extended control functionalities These features can be used to enhance performance:

Cogging Torque Compensation Removes ripples caused by magnetic interaction between motor rotor and stator.

Field Weakening The velocity range of electric motors can be extended by weakening the magnetic field of the rotor linearly over the speed range.

Anti-Windup Control An Anti-Windup algorithm is used to prevent the Integrator part to accumulate beyond acceptable boundaries.

Gain Scheduling Applies controller gains which will change dynamically regarding the current speed.

Motor Overload Protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating.

Velocity feed forward To increase the velocity of the robot's reaction, the target position signal can bypass the position controller.

Torque Offset A desired offset torque can be added to the reference torque.

Velocity Offset An additional velocity offset can be used to improve the positioning performance.

Filtering Digital filters can be used to remove unwanted frequencies such as noise or nonlinear effects.

Position and Velocity Feedback Low-pass filters

Torque loop input shaping filter

Command smoothing and Interpolation Successive setpoints can be smoothed, so that low command resolution and jitter have a reduced effect on the control.

Control supervision Allows to supervise the work of the control loops

Feed constant

Store home position

Distributed clocks Allows to synchronize the system time for enhanced performance.

OS Command Offers commands that can't be executed by single SDO operations.

Quick stop

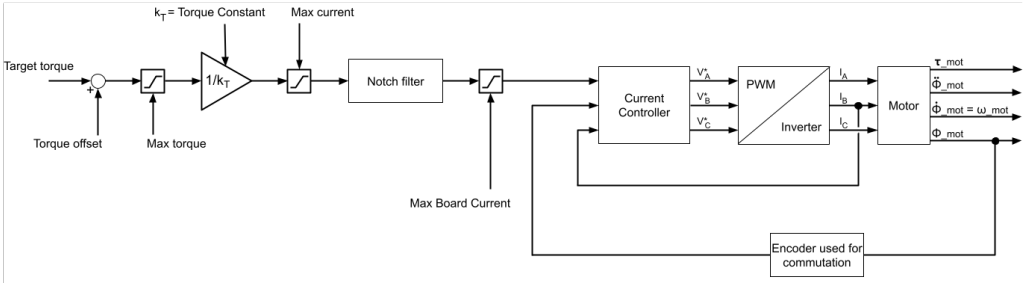
Control Loops

- Current and Torque Control Loop
- Velocity Control Loop
- Position Control Loop
- Dual Loop Cascaded Postion Control

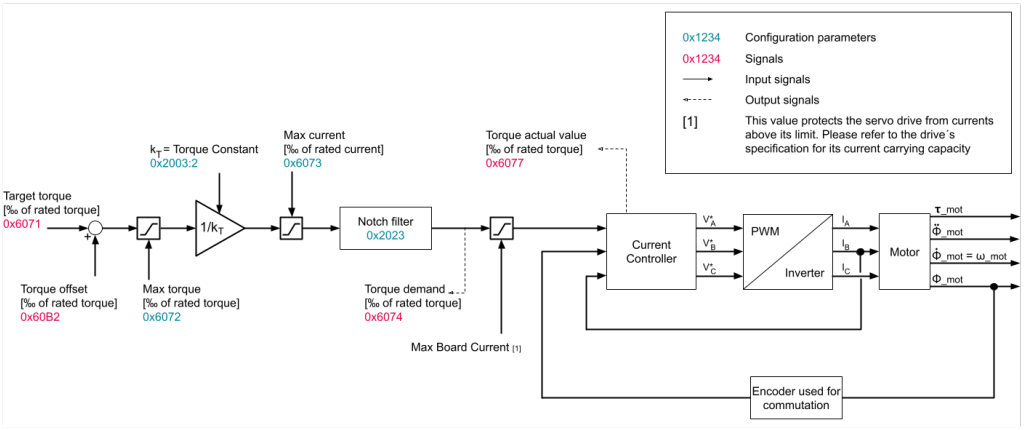
Current and Torque Control Loop

Torque control loop

General view



Detail view



> Note:
>
> The master commands a torque but internally it is computed as current. In most cases, as the position or velocity is the controlled value, the current measurement is sufficient. If a precise torque is required by the application, it should be verified with a torque sensor.

General Scope of the Torque Controller

The task of the torque controller is to make sure that the torque currently produced by the motor corresponds to the desired torque. This makes it one of the core algorithms of servo drives. It is used whenever a motor is driven.

In general, the RMS value of motor phase currents are proportional with the generated torque. The ratio between these two is known as **torque constant k_T** (as defined in subitem 0x2003:2).

The torque control loop is executed at a frequency of 16 kHz (as of firmware v4.2.0).

Reference Torque Generation

The reference torque can be updated in every execution step. In cyclic synchronous torque mode, the reference torque is transferred directly from the EtherCAT master to the torque controller via the PDO (object 0x6071). In other modes, a position or velocity controller is running on the SOMANET Drive and acts as higher level control, so their outputs act as reference for the torque controller. The offset torque 0x60B2 is also part of the reference torque as described in Torque Offset.

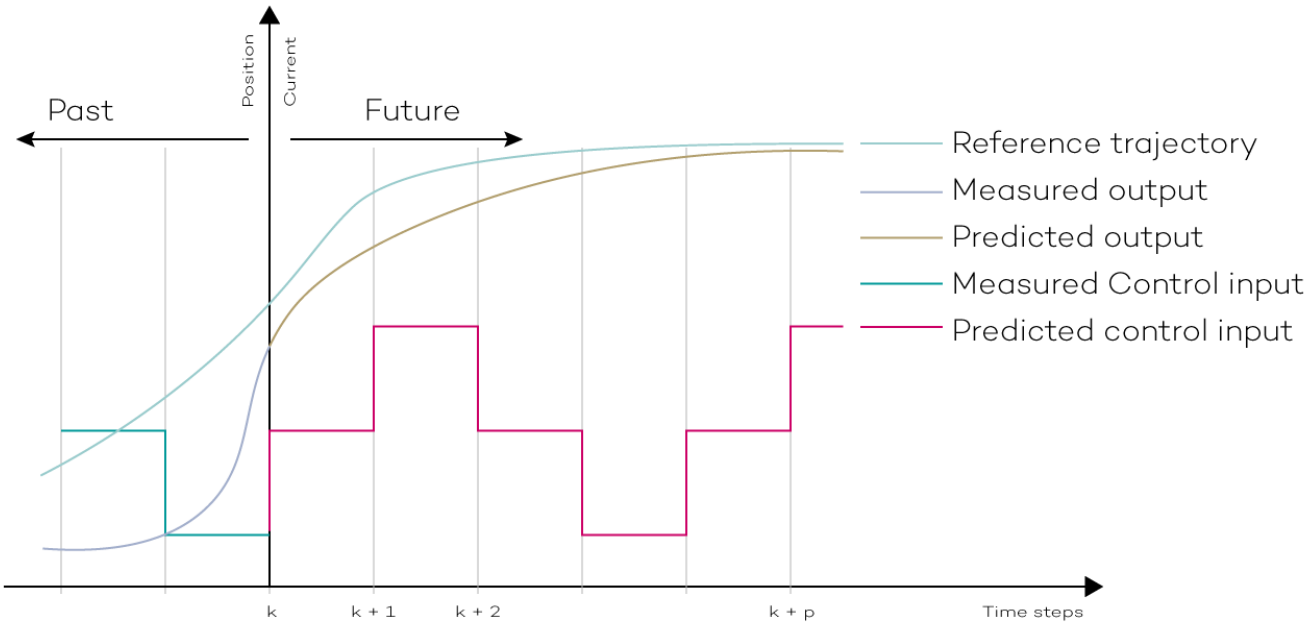
Model-Predictive Deadbeat Current Controller

The current controllers used in SOMANET are using a Model-Predictive Deadbeat Field-oriented Controller plus an Integrator.

The purpose of the model-predictive current controller (block “Current Controller” in the block diagram) is to make the actual torque-generating current match the requested torque-generating current. Just as a regular PID current controller, the model-predictive current controller tries to remove the current error.

In Model-Predictive Control, the model of the system is known to the controller. This enables the controller to “predict” the optimal control value for the next time step taking into account the value from the actual time slot.

While a conventional PI current controller would iteratively change the PWM duty cycle to reach a certain current, the model predictive controller can predict the required PWM duty cycle and set it right away during one time step.



Instead of a PID structure, it internally uses a model-based approach to determine the required voltages at motor terminals. The underlying equations allow predicting the current over one sampling period and calculate the required voltages so that by the end of the following voltage application period, the current errors will be suppressed to 0.

As a base for these calculations, the motor parameters are used. This controller structure can theoretically suppress the error values of motor currents to 0 in only one control execution period. However, it is sensitive to sensor noise and also depends on the model accuracy.

Functional Description and Controller Structure

The following figure represents the structure of the torque controller. The implemented torque control algorithm is based on field oriented control (FOC).

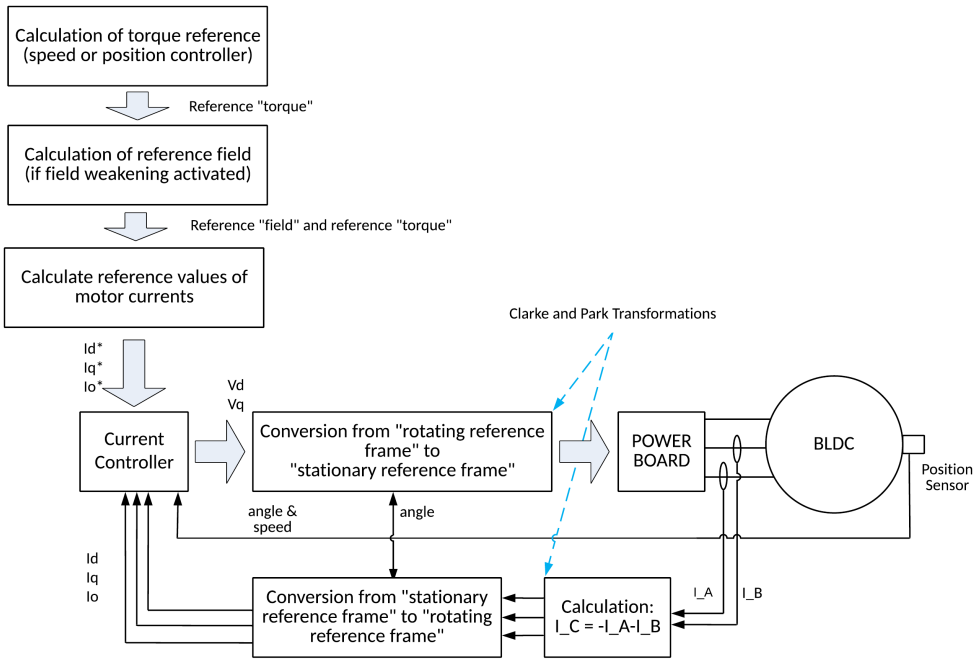


Figure 3: Torque loop structure

Tuning of the current controller

The controller can be fine-tuned by directly setting two parameters:

- Settling time (subitem 0x2010:10)
- Damping ratio (subitem 0x2010:11)

Settling time is entered in [µs], the default value 1000 µs.

Damping ratio is entered in [per mill], the default value is 2000 ‰.

Use case:

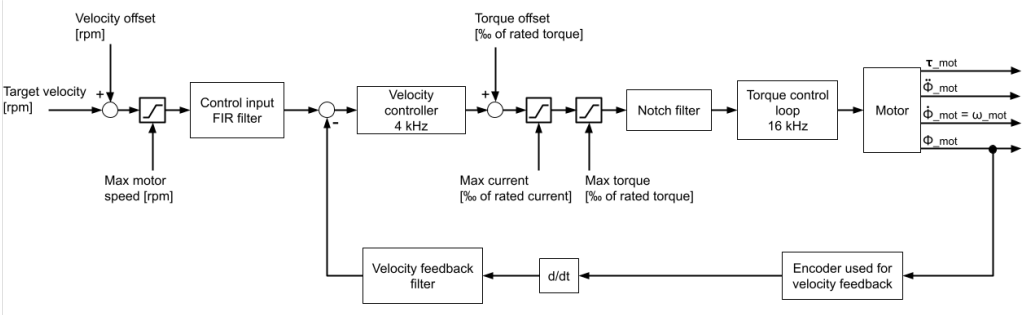
For a damping ratio of 0.7, the entered value should be 700 [per mill].

Parameters

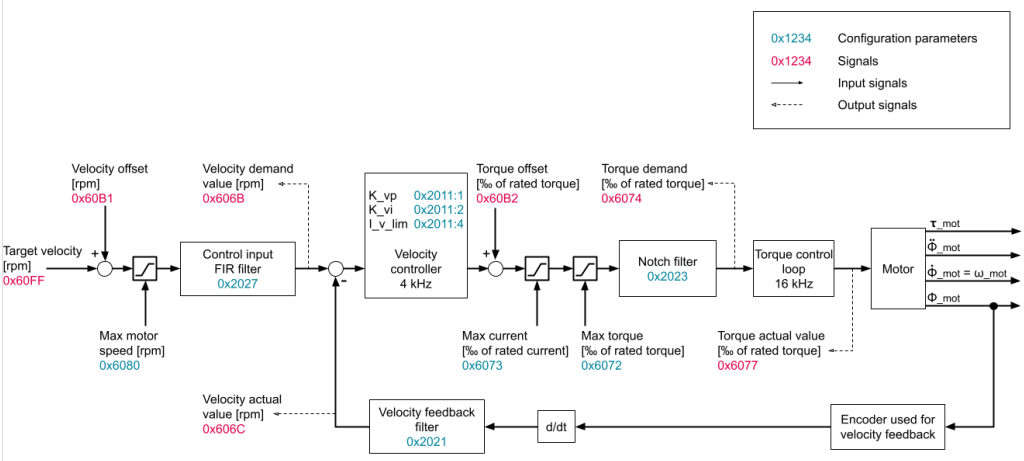
Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller

Velocity Control Loop

General view



Detailled view



The internal structure of the velocity controller is a PID feedback structure, along with anti-windup on the integrator and low-pass filter on the velocity feedback signal to deal with the sensor noise. **It is recommended to use this controller only as a PI controller (leave the D gain set to 0).**

On top of the desired velocity input, the velocity controller also accepts an additional velocity off-set and additionaly, a torque offset can be added to the control loop to improve the control

performance. This dynamically changeable torque value will be added to the torque value generated by the velocity PI controller.

Although the cascaded position control loop also includes a velocity feedback loop, the velocity controller described here and especially its controller gains are not used as part of any position controller. This means the controller gains for the velocity controller described here (objects 0x2011:1 … 0x2011:4) and the velocity cascade being part of the position controller (objects 0x2012:5 … 0x2012:8) are set independently.

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2011	Velocity Controller	Values Kp, Ki, Kd and Controller Integral Limit for Velocity Controller

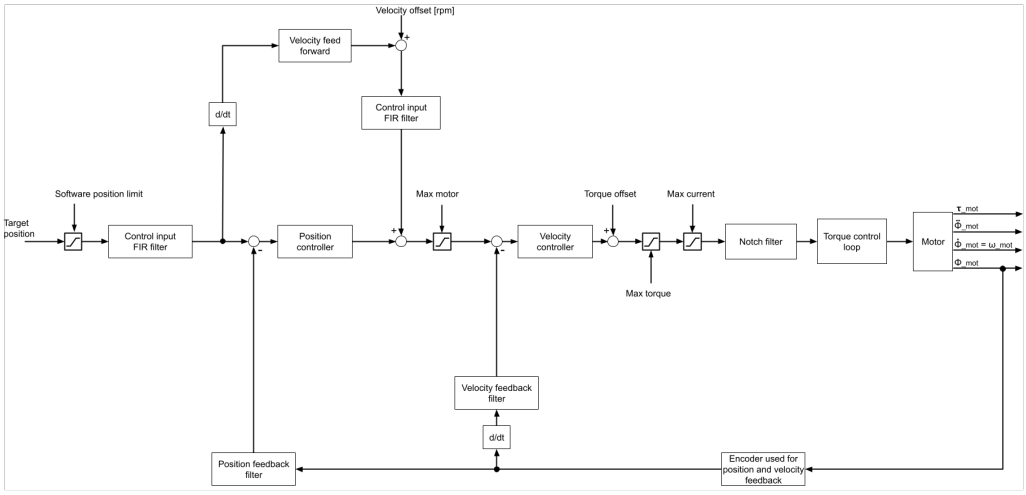
Position Control Loops

There are two implementations available for position control:

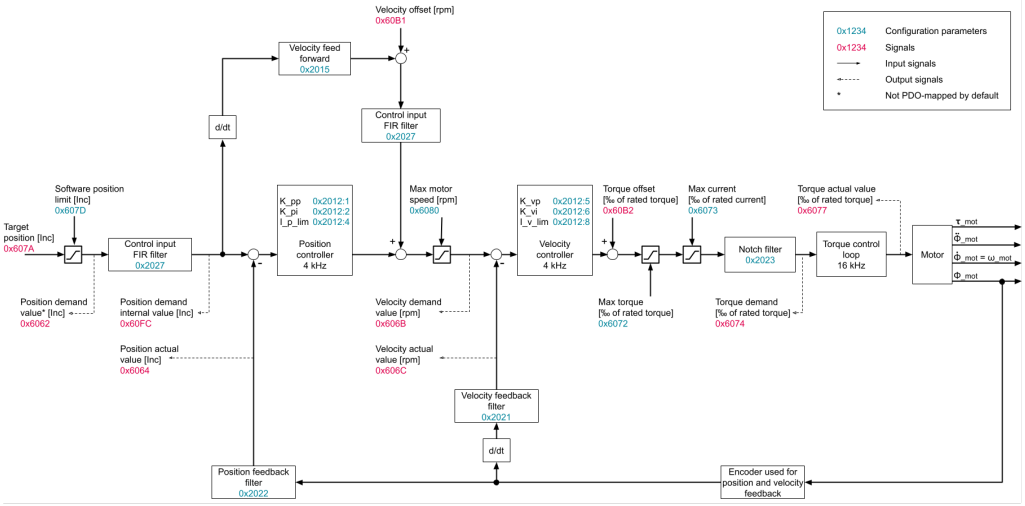
- Cascaded PID controller
- Simple PID controller

Cascaded PID Controller

General view



Detailled view



Note:

The specified units of the PDOs are valid only for the default unit configuration.

If user defined velocity (e.g. mRPM, rad/s) or position units are configured, they will be used instead. Additionally, in this particular configuration, the following objects: 0x606C Velocity actual value, 0x60B1 Velocity offset and 0x606B Velocity demand value are defined in the gearbox output reference frame, not in the motor frame.

See more about the velocity reference frame in the Migration Guide

Attention:

The velocity FIR filter is before velocity feed forward until firmware 5.0.3! Only in the soon-to-be-released firmware 5.0.4 it will be at the shown position.

The cascaded position controller consists of two PID feedback loops. The outer loop is used to compare the target position (current position setpoint) with the actual position value. From this difference, the PID controller computes a target velocity. The inner loop will compare this target velocity with the currently measured velocity and feed this difference to another PID controller that generates a target torque from this. This target torque is then controlled by the torque / current control loop described in the section on Torque Control. The overall control structure including the

underlying torque control loop, the relevant update frequencies and the object numbers is shown in a simplified form in the figure above.

Although the two control loops offer a total of 8 controller parameters (P-gain, I-gain, D-gain and an integrator limit), it is not useful to use all of them at the same time. Generally, none of the D-gains should be used in the cascaded controller and the two I-gains should be used only alternatively, not together. This means that the control structure is simplified to a combination of one P controller and one PI controller: PI-P or P-PI.

Instead of the cascaded PID controller, OBLAC Drives allows the user to activate the simple PID position controller. This controller represents the simplest and most known form of position control. However, for productive use Synapticon recommends to use the cascaded controller. In the simple PID controller, one Proportional, Integral and Derivative feedback acts on the position itself and will directly generate the target torque / current.

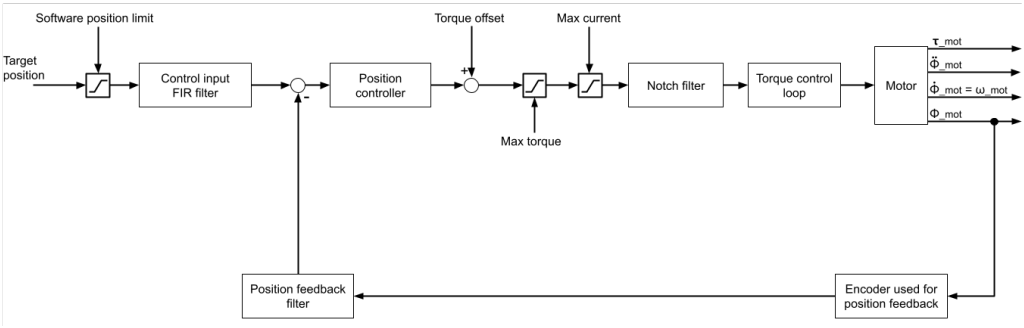
On top of the desired position input, both position control implementations also accept a desired 'offset torque', which is a dynamically changeable torque value that will be added to the torque value generated by the velocity PI controller. For further reference see Torque Offset.

Parameters

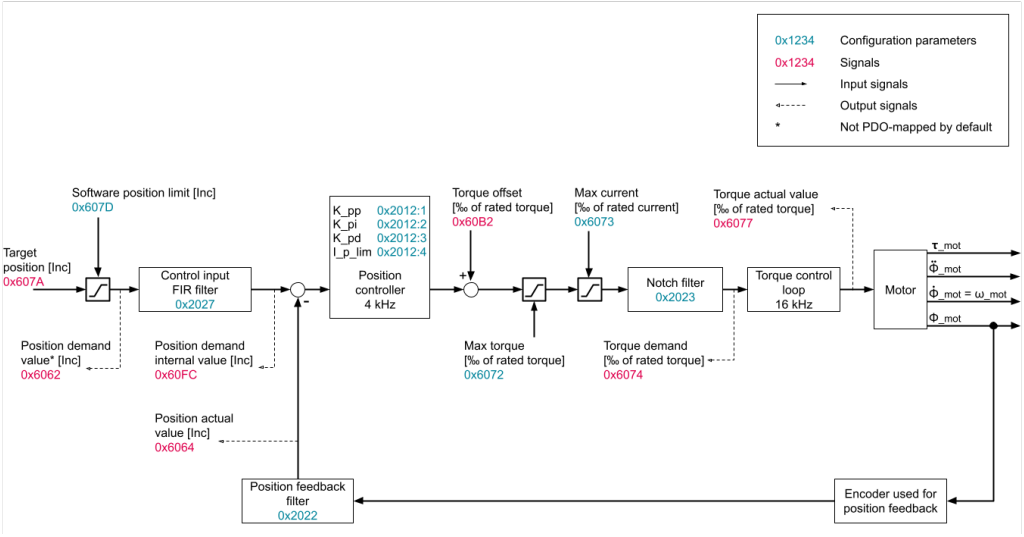
Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)

Simple PID Controller

General view



Detailled view



The cascaded PID controller is generally preferable over the simple PID controller. It is activated by default and apart from academic purposes there are few reasons to switch to the simple PID controller. However, the user can decide which position control structure is preferable for the particular application scenario.

Parameters

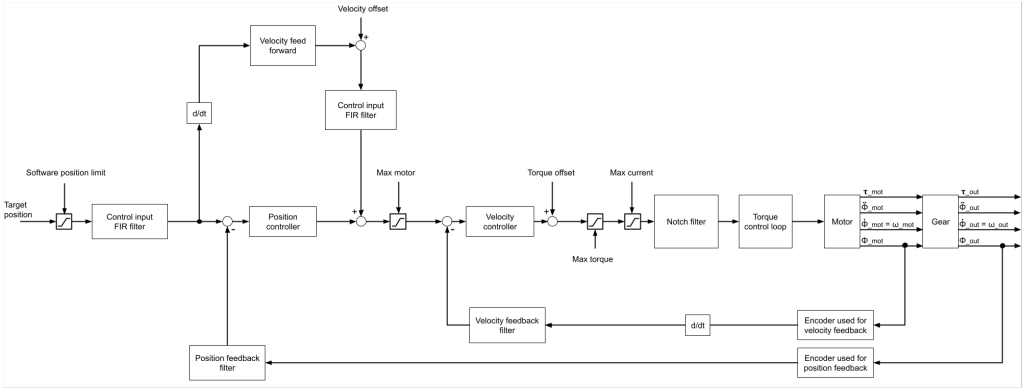
Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Simple PID Controller (Only subitems 1-4)

Dual Loop Cascaded Position Control

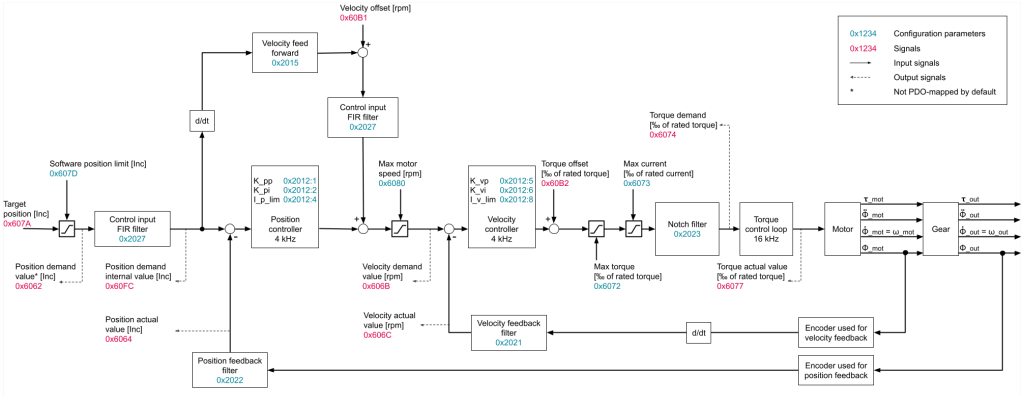
Overview

Using a gear can cause inaccuracy when only the shaft is controlled.
A second encoder can be attached to the output shaft after the gear for an optimized position control.
This section shows exemplary how the cascaded position controller can be configured for a dual encoder setup.

General view



Detailled view



Note:

The specified units of the PDOs are valid only for the default unit configuration.

If user defined velocity (e.g. mRPM, rad/s) or position units are configured, they will be used instead. Additionally, in this particular configuration, the following objects: 0x606C Velocity actual value, 0x60B1 Velocity offset and 0x606B Velocity demand value are defined in the gearbox output reference frame, not in the motor frame.

See more about the velocity reference frame in the Migration Guide

Attention:

The velocity FIR filter is before velocity feed forward until firmware 5.0.3! Only in the soon-to-be-released firmware 5.0.4 it will be at the shown position.

Usage

- Set the first encoder at the input shaft to *Commutation* and *Velocity*
- Set the second encoder at the output shaft to *Position*
- Ensure that the gear ratio is set correctly in object 0x6091.
- Set the encoder installed on the motor side (gear input shaft) to Commutation and Velocity functions.
- Set the encoder at the gear output shaft to Position function.
- More details on encoder configuration can be found here.
- Ensure that the gear ratio is set correctly in object 0x6091.

Parameters

Object ID	Name	Description
0x6091	Gear Ratio	Object for indicating the revolutions of motor and shaft
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)

Modes of Operation

SOMANET supports different modes of operation that either utilize a trajectory controller on the slave (e.g. profile modes) or get their trajectories from a master and cyclically decrease the difference between target and actual values (cyclic modes). Additionally, Homing modes are available to move the axis to a specific position.

Cyclic modes: In cyclic synchronous modes the trajectory must be computed by the master.

Cyclic synchronous torque Mode (CST): In torque control mode the master sends desired torques to the drive.

Cyclic synchronous velocity Mode (CSV): In velocity control modes the master sends desired velocities to the drive.

Cyclic synchronous position Mode (CSP): In position control modes the master sends desired positions to the drive.

Profile modes: In profile modes, the trajectory is computed on the slave.

Profile position mode: In profile position mode the trajectory to a target point is generated on the drive.

Profile velocity mode: In profile velocity mode the motor will rotate at a target velocity.

Profile torque mode: In profile torque mode the drive sends a target torque to the motor.

Homing modes: Homing can be used to center an axis on its current position or to move to an index pulse of the encoder.

Diagnostics opmode: Used for triggering diagnostic OS commands.

Switching modes: The modes are called by using Object 0x6060.

Cyclic modes

- Cyclic Synchronous Torque Control (CST) In torque control mode the EtherCAT master sends desired torques to the drive.
- Cyclic Synchronous Velocity Control (CSV) In velocity control modes the EtherCAT master sends desired velocities to the drive.
- Cyclic Synchronous Position Control (CSP) In position control modes the EtherCAT master sends desired positions to the drive.

In cyclic synchronous modes reference values are sent and updated with each EtherCAT communication frame, so a reference signal has to be sent with up to 4 kHz. As an example, to rotate the motor shaft from position A to position B in cyclic synchronous position mode within one second, the following information has to be sent to the drive:

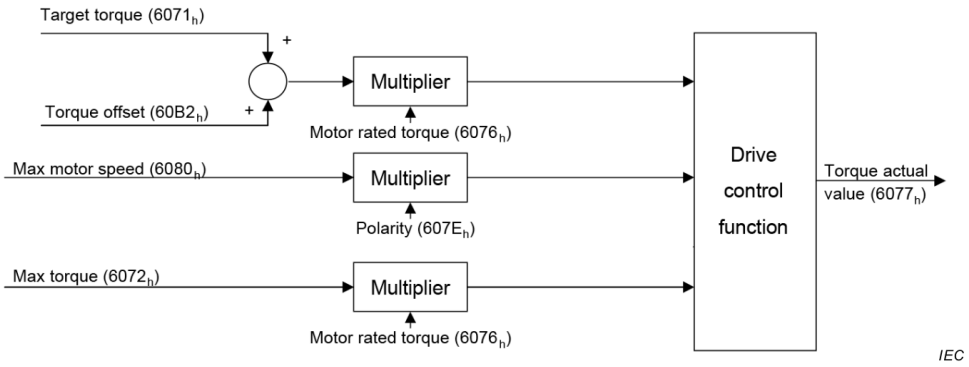
- position A in the beginning
- 3998 intermediate positions (ideally at 4 kHz)
- position B in the end

Cyclic Synchronous Torque Mode

In this mode, the trajectory generator is located in the control device, not in the servo drive. In a cyclic synchronous manner, it provides a target torque to the drive device, which performs torque control. Optionally, an additive torque value can be provided by the control system in order to allow two instances to set up the torque.

The cyclic synchronous torque mode covers the following sub-functions:

- demand value input;
- torque capture;
- torque control function with appropriate input and output signals;
- limitation of torque demand.



The figure above shows the inputs and outputs of the torque control function. The input (from the control function point of view) are the target torque and optionally a torque offset which is added to the target torque to allow two instances to set up the torque.

The torque can be limited in object 0x6072 Max torque.

The torque actual value is used as mandatory output to the control device.

Operation Mode dependent Bits of the Statusword

This mode uses some bits of the statusword for mode specific purposes which are indicated in yellow. The figure shows the structure of the statusword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-13	12	11	10	9-8	7-0
N/A	Target Torque ignored	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
    - Max torque 0x6072 limits Target torque 0x6071 and Torque offset 0x60B2.
- **Target reached:** The bit is set if the actual torque stays in the window of target torque ± torque window value for a duration of torque window time.
- **Target torque ignored:** This bit is set if Target torque 0x6071 is equal to Torque demand 0x6074.

  It means that the user command is passed directly through as an input of the torque controller. However, if Target torque is limited by Max torque, the limited value will be used as an input of the torque controller and the bit will be cleared.

  > Note:
  >
  > 0 = Target torque ignored
  >
  > 1 = Target torque shall be used as input

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x6071	Target torque	Use this object to directly select a Target torque
0x60B2	Torque Offset	This object provides the offset for the torque value

Cyclic Synchronous Velocity Mode

The velocity control modes allow the EtherCAT master to send desired velocities to the SOMANET drive. The drive will make sure these velocity values are reached by the motor (if the motor is physically able to do so).

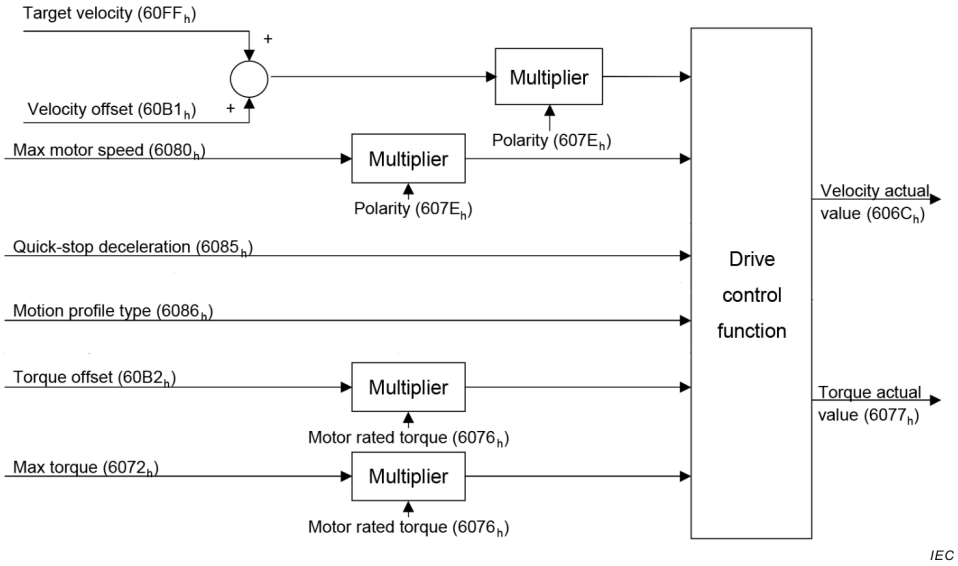
In this mode, the trajectory generator is located in the control device, not in the servo drive. In a cyclic synchronous manner, it provides a target velocity to the drive device, which performs velocity control and torque control. If desired, the position control loop may be closed over the communication system. Optionally, a velocity offset and a torque offset may be provided by the control system in order to allow a second source for velocity and/or a torque feed forward.

Position sensors are supported and the velocity may be calculated by taking the derivative.

The cyclic synchronous velocity mode covers the following sub-functions:

- demand value input;
- velocity capture using position sensor;
- velocity control function with appropriate input and output signals.

The behavior of the control function is influenced by control parameters such as limit functions, which are externally applicable.



The figure above shows the inputs and outputs of the drive control function. The input (from the control device point of view) are the target velocity and optionally, a velocity offset (to be added to the target velocity) as well as a torque offset. The drive control function in the diagram contains both the velocity controller and the torque controller, where the output of the velocity control loop is used as an input for the torque control loop.

The drive device supports limitation of motor speed and a quick stop function for emergency reasons.

The time period between two updates of the target velocity and/or additive velocity can be interpolated for smooth trajectories.

The velocity actual value is used as mandatory output to the control device. Another output is the torque actual value.

All values are in rpm by default and can be converted to user defined velocity units if needed.

Operation Mode dependent Bits of the Statusword

This mode uses some bits of the statusword for mode specific purposes which are indicated in yellow. The figure shows the structure of the statusword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-13	12	11	10	9-8	7-0
N/A	Target velocity ignored	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
    - Max. motor speed 0x6080 limits Target Velocity 0x60FF and Velocity Offset 0x60B1.
- **Target reached:** The bit is set if the actual velocity stays in the window of target velocity ± velocity window for a duration of Velocity window time.
- **Target velocity ignored:** This bit is set if Target velocity 0x60FF is equal to Velocity demand value 0x606B. It means that the user command is passed directly through as an input of the velocity PID controller.
- However, if Target velocity is limited by Max. motor speed, the limited value will be used as an input of the PID controller and the bit will be cleared.

> Note:
>
> 0 = Target velocity ignored
>
> 1 = Target velocity shall be used as input

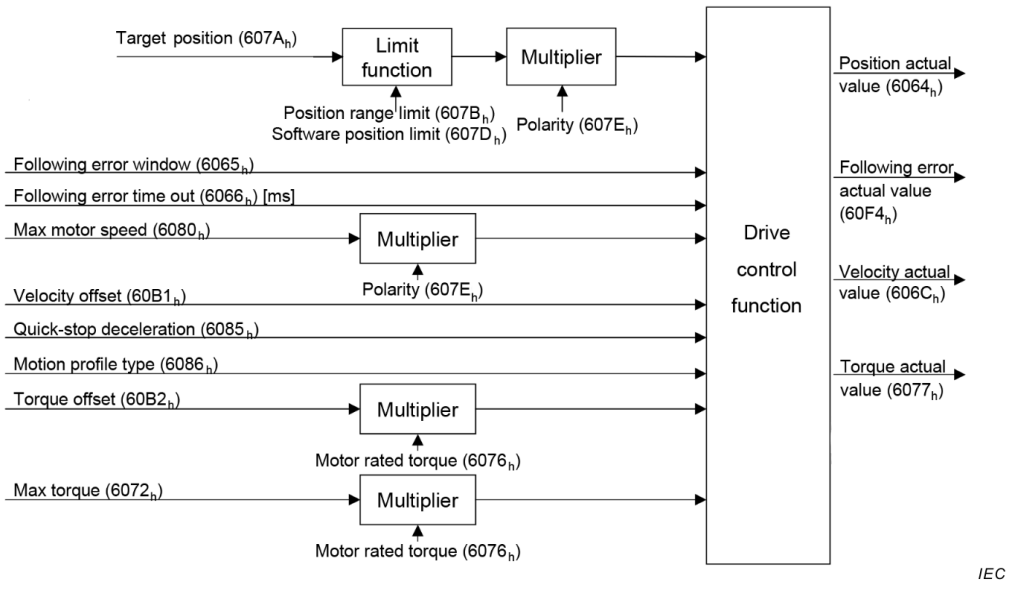
Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2011	Velocity Controller	Values Kp, Ki, Kd and Controller Integral Limit for Velocity Controller
0x60FF	Target velocity	Use this object to directly select a Target velocity

Cyclic Synchronous Position Mode

The position control modes allow the EtherCAT master to send desired positions to the servo drive. The drive will make sure these position values are reached by the motor (if the motor is physically able to do so).

In this mode, the trajectory generator is located in the control device, not in the servo drive. In a cyclic synchronous manner, it provides a target position to the drive device, which performs position control, velocity control and torque control.



The figure above shows the inputs and outputs of the drive control function. The input values (from the control function point of view) are the target position as well as an optional velocity offset and an optional torque offset used for feed forward control. Especially in a cascaded control structure, where a position control is followed by a velocity or torque control, the output of the position

control loop is used as an input for further calculation in the drive device. Limit functions can be used to restrict the range of values to avoid unintended positions. The drive device monitors the following error.

The target position is interpreted as absolute value. The position actual value is used as mandatory output to the control device. Further outputs are the velocity actual value and torque actual value. The following error actual value is used as an additional parameter. See Control supervision for details.

Three control loops are implemented for position control:

- Cascaded PID control
- Simple PID control
- Dual Encoder Cascaded Position Control

Operation Mode dependent Bits of the Statusword

This mode uses some bits of the statusword for mode specific purposes which are indicated in yellow. The figure shows the structure of the statusword. For the general structure and usage, please refer to our Application note on Status- and controlword.

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-13	12	11	10	9-0
N/A	Target position ignored	Internal limit active	Target reached	basic

- **Internal limit active:**
    - Max. motor speed 0x6080 limits Velocity demand value 0x606B and Profile velocity 0x6081.
    - Software position limit 0x607D limits Position demand value 0x6062.
- **Target reached:** the bit is set to 1 if the actual position stays in the window of the target position ± position window for a duration of position window time.

- **Target position ignored:** This bit is set if Target position (0x607a) is equal to Position demand internal value (0x60FC). It means that the user command is passed directly through as an input of the position PID controller.

  However, if the Target position is limited by Software position limits, the limited value will be used as an input of the PID controller and the bit will be cleared.

> Note:
>
> 0 = Target position ignored
>
> 1 = Target position shall be used as input

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values Kp, Ki and Kd for Torque Controller
0x2012	Position Controller	Values Kp, Ki and Kd for Cascaded Position Controller (Both loops: Position and Velocity)
0x607A	Target position	Use this object to directly select a Target position
0x60B1	Velocity offset	Sets a feed-forward velocity as input to the velocity controller
0x60B2	Torque offset	Sets a feed-forward torque as input to the velocity controller

Profile modes

SOMANET allows to generate a linear profile for position and velocity loops.

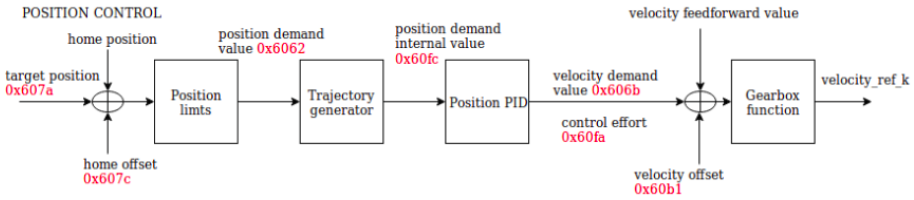
Profile position mode: In profile position mode the trajectory to a target point is generated on the drive.

Profile velocity mode: In profile velocity mode the motor will rotate at a target velocity.

Profile torque mode: In profile torque mode the drive sends a target torque to the motor.

Profile position mode

In Profile Position Mode a target position is applied to the trajectory generator. It is generating a position demand value for the position control loop taking into account the position limits. The trajectory generator computes the position demand internal value depending on the configured speed, starting acceleration and braking deceleration.



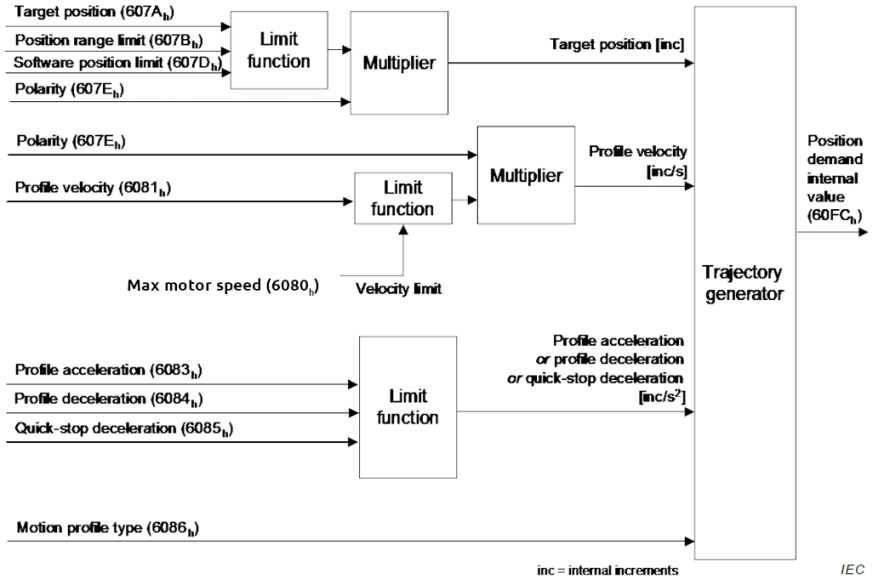
Objects configuration

The following objects need to be configured for the position profile:

- 0x6081 - Profile Velocity
- 0x6083 - Profile acceleration
- 0x6084 - Profile deceleration
- 0x6085 - Quick Stop deceleration

> Note:
>
> Profile velocity is limited by the Object:
>
> Max motor speed 0x6080

Trajectory generator

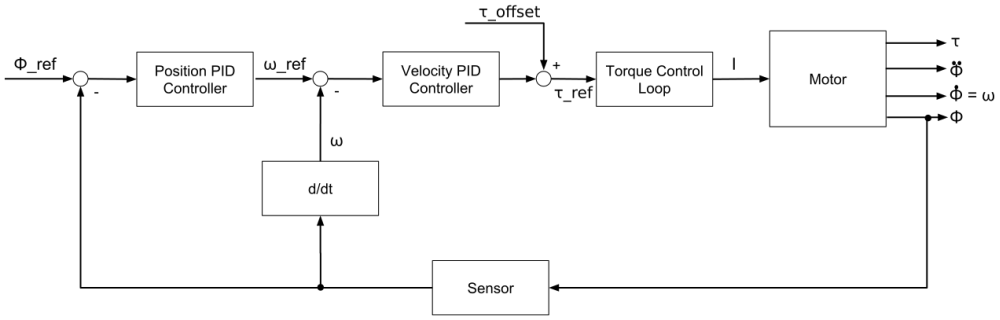


The trajectory generator receives the Target position 0x607A, taking into account the Position range limit 0x607B, the Software position limit 0x607D, as well as the Polarity 0x607E.

> Note:
>
> Target position is limited by the Objects:
>
> - Position range limit 0x607B
>
> - Software position limit 0x607D

The output of the trajectory generator is the position demand internal value for the position PID.

General view



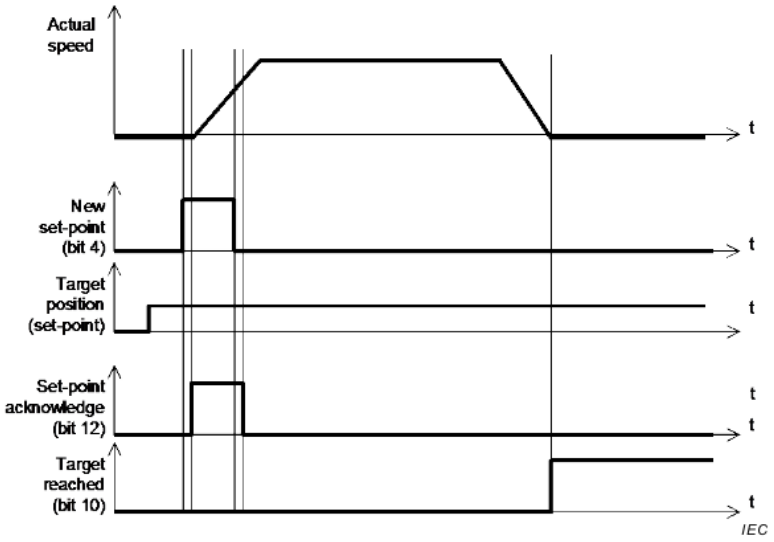
Enable Profile position mode

To enable the mode, the value 1 (0001h) must be entered in object Modes of operation 0x6060. The object Op mode display 0x6061 can be used to check if the op mode has been set correctly.

Setting set-points

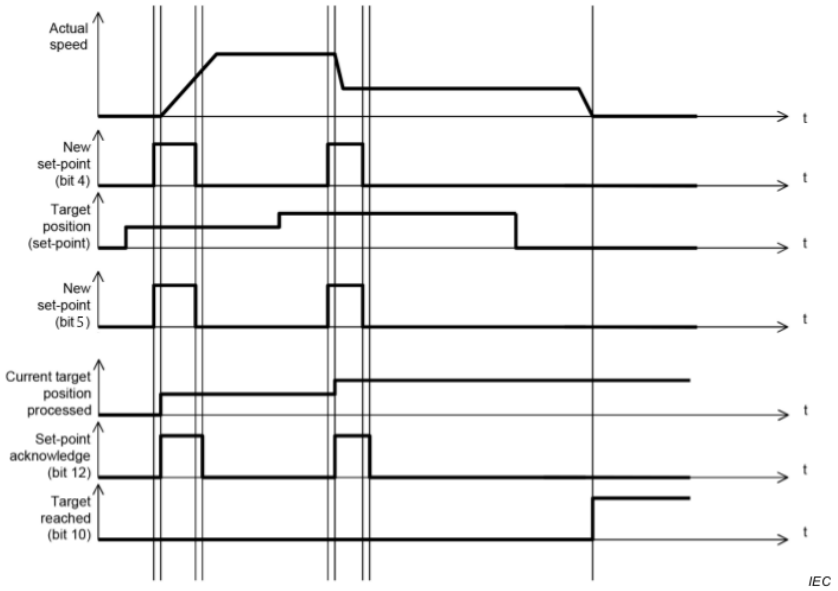
The setting of set-points is controlled by the timing of the *new set-point bit* in the controlword as well as the *set-point acknowledge bit* in the statusword.

After a set-point is applied to the drive device, the control device signals that the set-point is valid by a rising the edge of the *new set-point bit* in the controlword. The drive device sets the *set-point acknowledge bit* in the statusword to 1, afterwards the drive device signals with the *set-point acknowledge bit* set to 0 its ability to accept new set-points.



In order to execute the new set-point (new target value), bit 4 (*new set-point*) of the controlword needs to be changed from 0 to 1 (the profile reacts on the rising edge of the bit). The drive will respond with the rising edge of bit 12 (*set-point acknowledge*) of the statusword. The Single set-point method is used, which means that bit 5 of the controlword (*Change set immediately*) has to be set to 1.

> Note:
>
> The drive won't wait for the previous set-point to finish.
>
> To execute the new set-point during the execution of the previous one, the **rising edge of the `new set-point` bit needs to be generated again.**



Operation Mode dependent Bits of Status- and Controlword

Controlword

The profile position mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

15-9	8	7-6	5	4	3-0
N/A	Halt	N/A	Change set imme-diately	New set-point	basic

- **New set-point:** On the rising edge, the drive will acknowledge the value of “target position” as a new set-point.
- **Change set immediately**: Our drive supports only a single-set-point method of handling. Therefore, this bit has to be 1 before setting a new-point.
- The new set-points will be executed immediately and won’t wait for the previous one to finish.

Attention:

This **bit 5 has to be set before or simultaneously with bit 4 `New set point`**. If this bit is not set, the profile command will not be executed!

- **Halt**: When the bit is activated, the drive will use the profile deceleration to put the motor's shaft into a standstill position.

Note:

The Halt bit is used to toggle the motion. When it is set to 1, the profile deceleration ramp will be used to reach 0 velocity. If it is set to 0 again, the demanded operation will be continued and the drive will accelerate using the profile acceleration ramp until the desired position has been reached.

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

Objects related to bits 12 and 13 of the statusword:

- 0x6065 - Following error window
- 0x6066 - Following error timeout
- 0x6067 - Position window
- 0x6068 - Position window time

15-14	13	12	11	10	9-8	7-0
N/A	Following error	Set-point acknowledge	Internal limit act-	Target reached	N/A	basic

15-14	13	12	11	10	9-8	7-0
			ive			

- **Internal limit active:**
    - Max. motor speed 0x6080 limits Velocity demand value 0x606B and Profile velocity 0x6081
    - Software position limit 0x607D limits Position demand value 0x6062.
    - Max torque 0x6072 limits Torque demand 0x6074.
- **Target reached:** the bit is set to 1 if the actual position stays in the window of the target position ± position window for a duration of position window time.
- **Set-point acknowledge:** Is set to 1 when the drive acknowledges the rising edge on the *new-set point bit* of the controlword.
- **Following error**: the bit is set to 1 If the actual position stays in the window of position demand value ± following error window for a duration of following error time out.
- For more details check the object Following error actual value 0x60F4.

> Note:
>
> If the position window is 0xFFFFFFFF position window monitoring will be switched off and the target reached bit will always be 0.

Example: Using the position profile

- set the profile objects:
    - Profile Velocity 0x6081
    - Profile acceleration 0x6083
    - Profile deceleration 0x6084
    - Quick Stop deceleration 0x6085
- set mode of operation to 1

- Set target position
- Switch the drive to “Operation Enabled”
- Set bit 5 of the controlword:
- Set bit 4 of the controlword:
    - The drive will acknowledge the new target position on the rising edge of bit 4 by activating the statusword bit 12 *Set-point acknowledge* for several cycles.
    - The motor will start rotating immediately.

> Note:
> Setting the bits 4 and 5 of the controlword has to be done in the same order shown above or simultaneously.

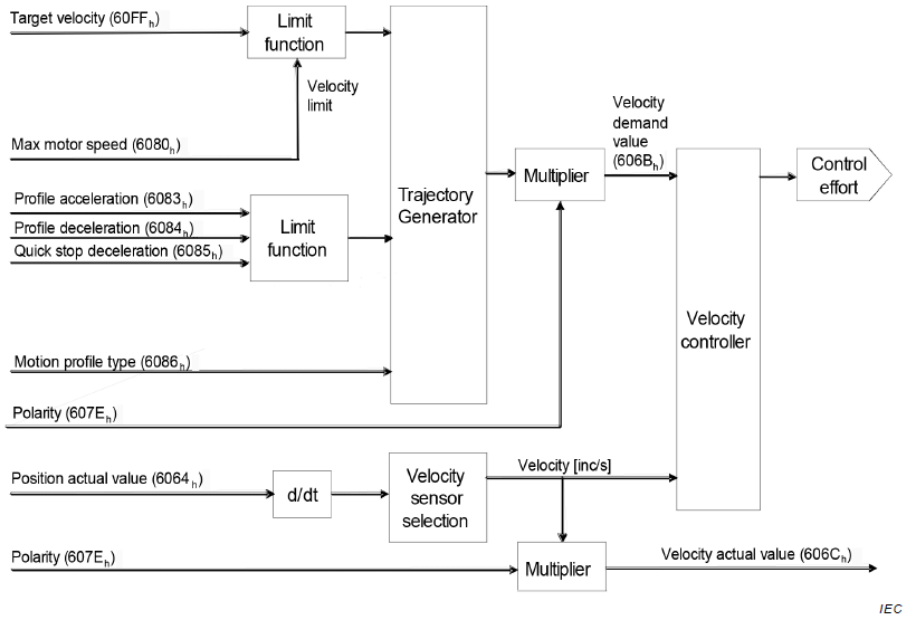
Parameters

Object ID	Name	Description
0x6081	Profile Velocity	
0x6083	Profile accel-eration	
0x6084	Profile decel-eration	
0x6085	Quick Stop decel-eration	
0x6080	Max motor speed	The maximum possible speed of the Motor according to the data sheet
0x6072	Max torque	
0x6074	Torque demand	

Object ID	Name	Description
0x607A	Target position	Use this object to directly select a Target position
0x607B	Position range limit	
0x607D	Software position limit	
0x607E	Polarity	
0x6060	Modes of operation	
0x6061	Op mode display	
0x6065	Following error window	
0x6066	Following error timeout	
0x60F4	Following error actual value	
0x6067	Position window	
0x6068	Position window time	
0x606B	Velocity demand value	
0x6062	Position demand value	

Profile velocity mode

In profile velocity mode the drive accelerates until the demand velocity value has been reached and keeps this velocity.



Objects configuration

The following objects need to be configured for the velocity profile:

- 0x60FF - Target velocity
- 0x6083 - Profile acceleration
- 0x6084 - Profile deceleration
- 0x6085 - Quick stop deceleration
- 0x607E * - Polarity
- 0x6080 - Max motor speed

* This object is automatically detected and configured by Commutation Offset Detection.

> Note:
>
> Target velocity is limited by Object:
>
> Max motor speed 0x6080

The motion in Profile velocity mode is toggled by the halt bit (8) in the controlword (object 0x6040):

- Value 0: Motion is executed, the drives accelerates (with the specified Profile acceleration) until either
    1. the velocity is reached or
    2. halt bit is set to 1
- Value 1: Motion is stopped, the drive decelerates (with the specified Profile deceleration) until either
    1. velocity = 0 or
    2. halt bit is reset to 0

> Attention:
>
> When the drive mode is set to Profile velocity mode and the halt bit is unset, the motor will start turning immediately.

Enable Profile velocity mode

To enable the mode, the value 3 (0003h) must be entered in object 0x6060. The Op mode display 0x6061 can be used to check if the op mode has been set correctly.

Operation Mode dependent Bits of Status- and Controlword

Controlword

The profile velocity mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

15-9	8	7-4	3-0
N/A	Halt	N/A	basic

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

Objects related to bits 10 and 12 of the statusword:

- 0x606D - Velocity window
- 0x606E - Velocity window time
- 0x606F - Velocity threshold
- 0x6070 - Velocity threshold time

5-13	12	11	10	9-8	7-0
N/A	Speed bit	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
    - Max. motor speed 0x6080 limits Velocity demand value 0x606B and Profile velocity 0x6081.
    - Max torque 0x6072 limits Torque demand 0x6074.
- **Target reached:** The bit is set if the actual velocity stays in the window of target velocity ± velocity window for a duration of *Velocity window time*.
- **Speedbit:**The speed bit is set to 1 when the actual velocity goes below the velocity threshold (meaning the motor is in standstill).

The bit will be 0 when the actual velocity exceeds the velocity threshold of the speedbit for a duration of *Velocity threshold time* (meaning the motor is rotating).

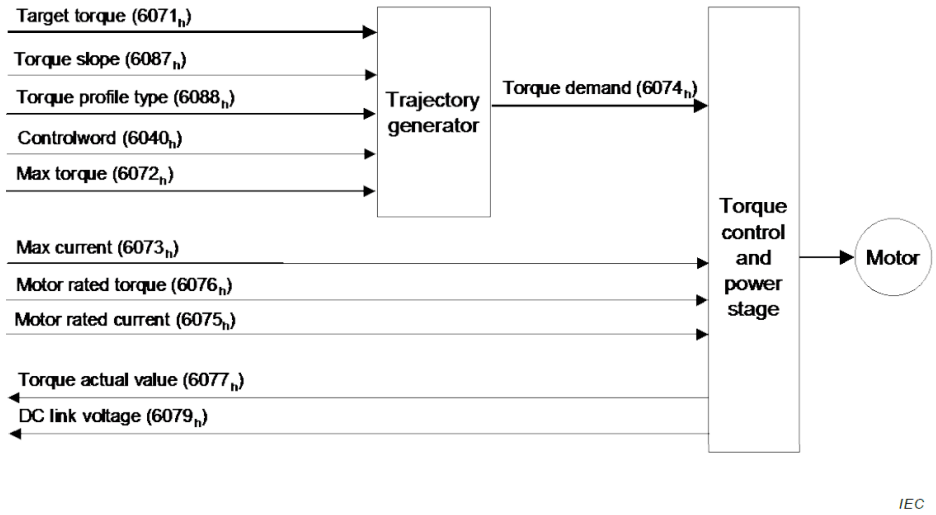
> Note:
> If the velocity window is 0xFFFF, velocity window monitoring will be switched off and the target reached bit will always be 0.

Parameters

Object ID	Name	Description
0x60FF	Target velocity	
0x6083	Profile acceleration	
0x6084	Profile deceleration	
0x6085	Quick stop deceleration	
0x607E	Polarity	
0x6080	Max motor speed	
0x606D	Velocity window	
0x606E	Velocity window time	
0x606F	Velocity threshold	
0x6070	Velocity threshold time	
0x6080	Max. motor speed	
0x606B	Velocity demand value	
0x6081	Profile velocity	
0x6072	Max torque	
0x6074	Torque demand	

Profile torque mode

In profile torque mode, a constant torque is generated according to the target torque and the defined torque slope (the rate of change of torque).



Output of the generator is the torque demand value (0x6074) which is processed by the torque controller.

Objects configuration

The following objects need to be configured for the torque profile:

- 0x6071 - Target torque
- 0x6087 - Torque slope
- 0x6085 - Quick Stop deceleration

> Note:
>
> 0x6071 Target torque is limited by the Object :
>
> Max torque 0x6072

Enable Profile torque mode

To enable the mode, the value 4 (0004h) must be entered in object 0x6060. The Op mode display 0x6061 can be used to check if the op mode has been set correctly.

> Note:
>
> Both enabling the halt bit and setting the target torque to zero will ramp down the torque

> applied to the motor according to the torque slope. At the end of the slope no torque will be applied to the motor, allowing the shaft to move freely.

Starting/Stopping Torque

To start and stop motion, toggle the controlword halt bit (bit 8). When the halt bit is set to 0, motion will start or continue; when the halt bit is set to 1, motion will stop. The bit can be toggled by writing 010Fh and 000Fh to controlword (object 0x6040).

Operation Mode dependent Bits of Status- and Controlword

Controlword

The torque position mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage, please refer to our Application note on Status- and Controlword.

15-9	8	7-4	3-0
N/A	Halt	N/A	basic

- **Halt:** Both enabling the halt bit or setting the target torque to zero will ramp down the torque applied to the motor according to the torque slope object (0x6087). At the end of the slope, no torque will be applied to the motor, allowing the shaft to move freely.

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information please refer to the section on Control Supervision.

15-12	11	10	9-8	7-0
N/A	Internal limit active	Target reached	N/A	basic

- **Internal limit active:**
    - Max torque 0x6072 limits Torque demand 0x6074, Target torque 0x6071 and Torque offset 0x60B2
- **Target reached:** the bit is set if the actual torque stays in the window of target torque ± torque window value for a duration of torque window time.

Objects related to the bit 10 of the statusword:

Torque window value 0x606D Torque window time 0x606E

Parameters

Object ID	Name	Description
0x6071	Target torque	
0x6087	Torque slope	
0x6085	Quick Stop deceleration	
0x6072	Max torque	
0x6074	Torque demand	
0x60B2	Torque offset	
0x606D	Torque window value	
0x606E	Torque window time	

Homing modes

Overview

Axes without an absolute encoder can use the homing function to restore the position to a previously defined location.

Homing methods

SOMANET supports the following methods as defined in CiA 402 profile specification (IEC61800-7-201:2015):

Method Number	Description
1	Homing on a negative limit switch and index pulse
2	Homing on a positive limit switch and index pulse
3	Homing on a positive home switch inactive and index pulse
4	Homing on a positive home switch active and index pulse
5	Homing on a negative home switch inactive and index pulse
6	Homing on a negative home switch active and index pulse
7	Homing on a home switch (inactive) and index pulse with positive initial direction. Reverse if positive limit switch is hit
8	Homing on a home switch (active) and index pulse with positive initial direction. Reverse if positive limit switch is hit
9	Homing on an opposite home switch (active) and index pulse with positive initial direction. Reverse if positive limit switch is hit
10	Homing on an opposite home switch (inactive) and index pulse with positive initial direction. Reverse if positive limit switch is hit
11	Homing on a home switch (inactive) and index pulse with negative initial direction. Reverse if negative limit switch is hit
12	Homing on a home switch (active) and index pulse with negative initial direction. Reverse if negative limit switch is hit
13	Homing on an opposite home switch (active) and index pulse with negative initial direction. Reverse if negative limit switch is hit
14	Homing on an opposite home switch (inactive) and index pulse with negative initial direction. Reverse if negative limit switch is hit

Method Number	Description
33	Homing on an index pulse with negative initial direction
34	Homing on an index pulse with positive initial direction
35/37 (identical)	Homing on the current position

There are also some manufacturer specific homing modes implemented:

Method Number	Description
-1	Homing on Negative End stop and Index Pulse.
-2	Homing on Positive End stop and Index Pulse.
-3 & -4	Similar to methods -1 and -2, but instead of Homing on Index pulse, homing is done at an offset from end stop.

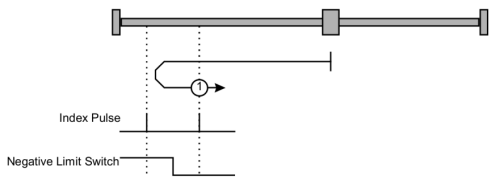
Homing methods 1-14 (Homing on limit switches)

In all switch based homing methods the axis moves to search the switch (Home Switch or Limit Switch).

The following images represent the direction of movement and the starting position of each homing method.

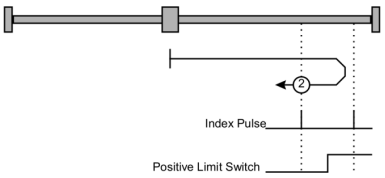
The respective method is indicated by the encircled number.

Homing Method 1 : Homing on a Negative Limit Switch and Index Pulse



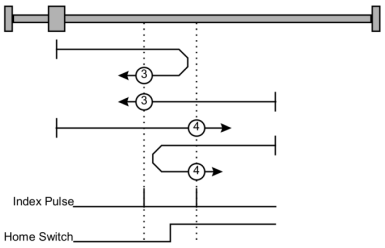
- The initial direction of movement shall be leftward if the negative limit switch is inactive.
- The home position is the first index to the right when the negative limit switch changes state.

Homing Method 2 : Homing on a Positive Limit Switch and Index Pulse



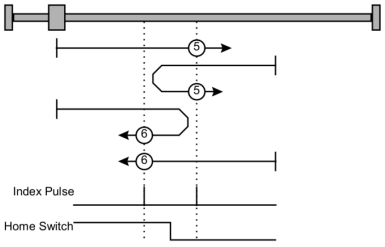
- The initial direction of movement shall be rightward if the positive limit switch is inactive.
- The home position is the first index to the left when the positive limit switch changes state.

Homing Method 3,4 : Homing on a Positive Home Switch and Index Pulse



- The initial direction of movement is dependent on the state of the home switch: leftward if positive home switch active and rightward if it’s inactive.
- Method 3: the home position is the first index to the left when the positive home switch changes state (home switch inactive).
- Method 4: the home position is the first index to the right when the positive home switch changes state (home switch active).

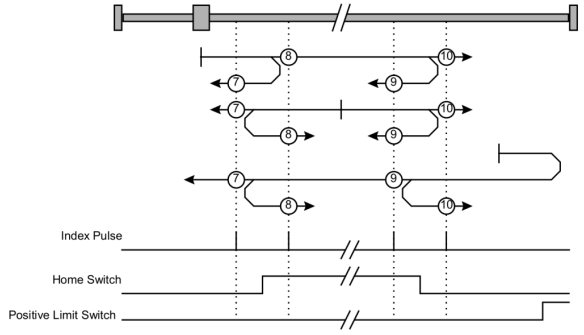
Homing Method 5,6 : Homing on a Negative Home Switch and Index Pulse



- The initial direction of movement is dependent on the state of the home switch: leftward if negative home switch inactive and rightward if it’s active.

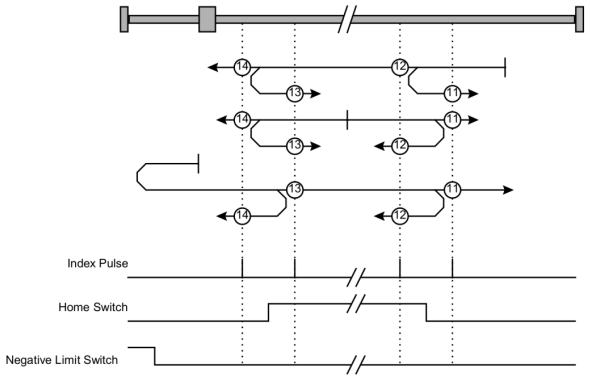
- Method 5: the home position is the first index to the right when the negative home switch changes state (home switch inactive).
- Method 6: the home position is the first index to the left when the negative home switch changes state (home switch active).

Homing Method 7 to 10 : Homing on a Home Switch and Index Pulse , Positive Limit Switch Required to revert direction



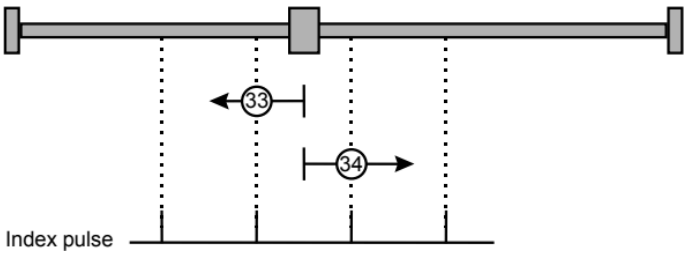
- The initial direction of the movement is rightward except if home switch is initially active.
- If the home switch is initially active then the direction of movement depends on the homing method.
- The direction will be reversed when encountering the positive limit switch.

Homing Method 11 to 14 : Homing on a Home Switch and Index Pulse , Negative Limit Switch Required to revert direction



- The initial direction of the movement is leftward except if home switch is initially active.
- The direction will be reversed when encountering the negative limit switch.
- If the home switch is initially active then the direction of movement depends on the homing method.

Homing method 33/34 (move to next or previous single-turn position or “index pulse”)



- Methods 33 and 34 can be used to center on the sensor’s reference position (in positive or negative direction).
- The shaft will rotate less than one rotation.

Homing method 35/37 (zero the current position)

- Methods 35 and 37 are used to define the current position as home position.

Note:

Homing method 35/37 can be performed in any operational state. All other homing methods require the drive to be in “operation enabled” state.

Use case:

Homing on current position while operation is not enabled

If the axis is moved to a desired homing position by external force (for example manually),

Homing Mode 35/37 can be used to determine the current position as home.

Use case:

Relative position command

The current position can be taken as the reference point.

Use case:

Homing on current position

When the motor is moved to the desired position in CSP the position can be taken as homing position.

Note:

Homing method 35/37 can be performed in any state, as the motor is not controlled there.

All other homing methods require the drive to be in “operation enabled” state.

Homing method -1,-2,-3 and -4 (Homing on mechanical endstop)

Methods -1,-2,-3 and -4 homes the axis near (or at) the mechanical end stop. For all these methods, first a holding torque is determined in the beginning of the homing routine. To calculate the holding torque, the drive holds the position of the axis for a duration of 500 ms just after the homing start-bit in controlword is set and the holding torque is determined by saving the torque demand value at the end of this time.

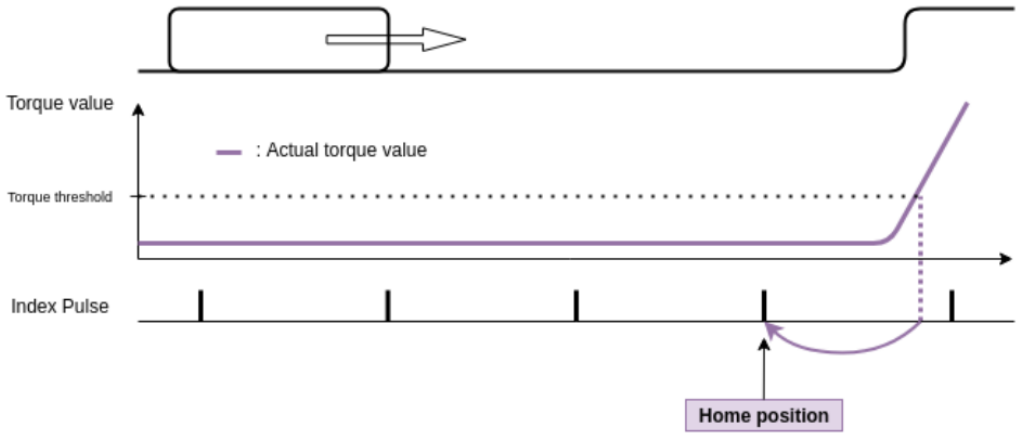
Method -1 moves the motor in the negative direction with velocity value defined in (0x6099:1). End stop is detected when

abs(demand_torque - holding_torque) >= torque threshold

The value of the torque threshold is defined in 0x2005:03. Upon detection of Mechanical endstop axis moves to the index pulse location in opposite direction with.the speed specified in object (0x6099:2). Axis is then homed at the index pulse.

Method -2 is similar to method -1, It just moves the axis in a positive direction towards the positive end stop and then home at the index pulse just before the endstop.

Methods -3 and -4 are similar to methods -1 and -2 respectively, but they work without index pulse. So upon detection of the Endstop, axis moves in the opposite direction by the offset position specified in 0x2005:04 and homes there. If value of 0x2005:04 is zero, then axis homes at the end stop.



Settings for homing mode

State for homing modes

All homing methods except for methods 35/37 require the drive to be in the “operation enabled” state. After the home position is found, the position controller stays active, holding the motor at that position (except for method 35/37).

Homing method

You can select the desired homing method in object 0x6098: Homing Method.

Mode	Description
-3 & -4	Similar to methods -1 and -2, but instead of Homing on Index pulse. Homing is done at an offset from end stop.
-2	Homing on Positive End stop and Index Pulse.
-1	Homing on Negative End stop and Index Pulse.
0	No homing method assigned (Default)
1	Homing on negative home switch and index pulse
2	Homing on positive home switch and index pulse
3,4	Homing on positive home switch and index pulse
5,6	Homing on negative home switch and index pulse
7 to 14	Homing on the home switch with an index pulse

Mode	Description
17, 18	Similar to method 1 & 2, but without index pulse. Homing is done at switch transition
19 to 22	Similar to methods 3 to 6, but without index pulse. Homing is done at switch transition
23 to 30	Similar to method 7 to 14, but without index pulse. Homing is done at switch transition
33	Homing on negative index pulse
34	Homing on positive index pulse
35	Homing on current position. (deprecated, please use method 37)
37	Homing on current position. (identical with method 35)

Homing speeds

Two homing speeds must be selected in object 0x6099: Homing Speed: in a typical cycle, the faster speed is used to find the home switch and the slower speed is used to find the index pulse.

> Note:
>
> Homing speeds are not required in method 35/37.

Homing acceleration

In object 0x609A: Homing Acceleration a value for acceleration during homing must be defined.

> Note:
>
> An acceleration is not needed in method 35/37.

If the drive is correctly homed, the bit 5 (PFV - Position Feedback Value) of the control word S-0-0135 is set. So for the master, the received 0 position is the real 0-position of the device (including gears on other conversion).

Procedure Commands for Homing

- S-0-0172 displacement to the reference system. Clear the PFV and tell the master that no homing position is set/active.
- S-0-0148 When the master sets and enables the Drive Controlled homing procedure command, the drive automatically activates the drive internal position control and accelerates to the homing velocity (S-0-0041) taking the Homing acceleration (S-0-0042) into account.
- S-0-0191 set the bit "position feedback value status" in S-0-0403.
- S-0-0447 set absolute position command

Setting digital I/Os for Homing

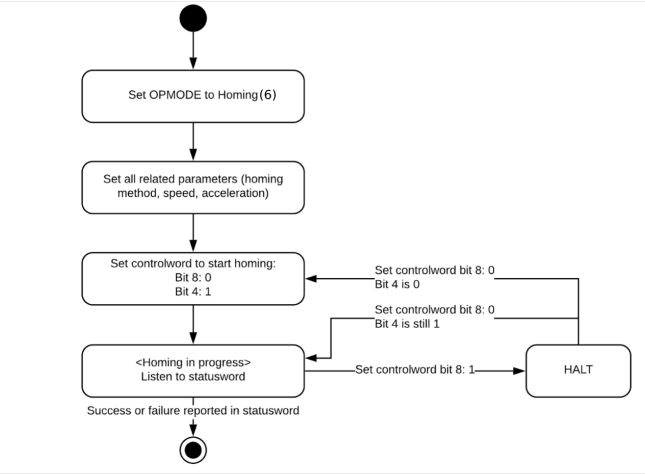
Object 0x2210 (GPIO Pin configuration) is used to configure the digital inputs for homing and limit switches. Switches can be configured in both active high and active low configuration. But only one GPIO can be assigned to one switch whether active high or active low. For example If Pin 3 is configured as Active High Limit Switch, then configuring any other pin as Limit Switch Active High or Active Low will raise the error IvldGpio (Invalid GPIO error) in the error report object.

Further details about configuring digital I/Os can be found in Digital GPIO

Object 0x60FD (Digital inputs) reports the state of the digital inputs which can be configured for limit switches, interlock, etc.

How to perform homing

The following diagram shows the steps that are necessary to perform the homing function.



After the home position is found, the position controller stays active, holding the motor at that position (except for method 35/37).

Using the Halt bit

The halt bit (bit 8 of the Controlword 0x6040 can be used to stop the drive without disabling the controller:

- Set the halt bit to 1 to stop the motion
- When the halt bit is reset to 0, the motion continues

Status of the homing process

Bits 10, 12, 13 of the Statusword 0x6041 are used to indicate the current state of the homing process:

- Bit 10: Target reached
- Bit 12: Homing attained
- Bit 13: Homing error

Bit 13	Bit 12	Bit 10	Status
0	0	0	Homing is performed
0	0	1	Homing is interrupted or not started
0	1	0	Homing confirmed, but target not yet reached
0	1	1	Homing completed
1	0	0	Error detected, motor still turning
1	0	1	Error during homing, motor at standstill

Hitting a limit switch in homing mode

Homing modes are meant to be used with homing switches whereas limit switches are considered as physical end points.

When a limit switch is activated in a homing mode other than method 1-14 (which rely on limit switches), this is considered a fault and the drive transitions to “Drive disabled” state.

Operation Mode dependent Bits of Status- and Controlword

Controlword

Homing mode uses some bits of the controlword for mode specific purposes which are indicated in yellow. The figure shows the structure of the controlword. For the general structure and usage,

please refer to our Application note on Status- and Controlword.

15-9	8	7-5	4	3-0
N/A	Halt	N/A	Homing operation start	basic

- **Halt:** If halt is triggered (Bit 8 of control word is set as 1) during homing, independant of the step the homing routine is in, Axis begins to ramp down to go to zero velocity. The axis stays at zero velocity as long as the halt remains active. When the halt bit is lifted, the homing routine is resumed from the same step it was in before halt.
- **Homing operation start:** set the homing operation start bit to 1 to start the homing procedure.

Statusword

The mode specific bits of the statusword can be used to monitor the performance of the operation. For further information, please refer to the section on Control Supervision.

15-14	13	12	11	10	9-8	7-0
N/A	Homing error	Homing attained	Internal limit active	Target reached	N/A	basic

Parameters

Object ID	Name	Description
0x6098	Homing Method	Specifies the homing method that shall be used
0x607C	Home Offset	Offset from user's zero position to the home position
0x6099	Homing Speeds	Speed with which to move during homing
0x609A	Homing Acceleration	Acceleration used during homing
0x6040	Controlword	Set bit 8 to 0 and bit 4 to 1 to start homing
0x6041	Statusword	Indicates the current state of the homing process
0x60FD	Digital inputs	Indicates the current state of the digital inputs
0x2210	GPIO Pin configuration	Configures the digital inputs for homing and limit switches
0x2005	Homing	Options configures different options for homing modes

Note:
Switches can be configured in both active high and active low configuration. But only one GPIO can be assigned to one switch, whether active high or active low. For example, if Pin 3 is configured as Active High Limit Switch, then configuring any other pin as Limit Switch Active High or Active Low will raise IvldGpio (Invalid GPIO error).

Diagnostics opmode

This operational mode allows triggering diagnostics OS commands. The following OS commands can only be executed when the drive is in this mode and when the CiA402 state is Operation

enabled:

- Command 4: Motor phase order detection
- Command 5: Commutation offset detection
- Command 6: Open phase detection
- Command 7: Pole pair detection
- Command 8: Phase resistance measurement
- Command 9: Phase inductance measurement
- Command 10: Torque constant measurement

To select Diagnostics opmode, the value of -2 must be written to the object 0x6060 Modes of operation. When this opmode is selected and the transition to Operation enabled is performed, the brake will only be disengaged if the encoder index needs to be found. Once this procedure is done, the brake will be engaged again and object 0x6041 Statusword will reflect that the drive is in Operation enabled state. When no OS command is running, the motor control is disabled.

> Attention:
> Always ensure that the brake is engaged or disengaged depending on the command that is executed:
> Command 4 rotates the rotor
> Command 5 rotates the rotor when the selected method is 0 or 1
> Command 6 does not need to rotate the rotor. In case a brake is attached, it may stay engaged during this command
> Command 7 rotates the rotor
> Command 8 rotates the rotor
> Command 9 rotates the rotor
> Command 10 rotates the rotor

For example, before running a diagnostics OS command that requires movement, the user should disengage the brake. When the OS command is finished the user should engage the

brake again, so that the load does not fall. An exception to this would be if the user is planning to execute another diagnostics OS command that requires movement, because in that case it could run it right after, so the motor control would be disabled just for a few milliseconds.

Selecting and switching the Modes

Selection of these modes is done by setting the object 0x6060 to the corresponding value as defined in the CiA 402 standard.

The modes are called by setting the object 0x6060 as follows:

Value for 0x6060	Control Mode
+1	Profile position mode
+3	Profile velocity mode
+4	Torque profile mode
+6	Homing mode
+8	Cyclic Syncronous Position Mode
+9	Cyclic Syncronous Velocity Mode
+10	Cyclic Syncronous Torque Mode

Dynamic Op-Mode switching

The modes can be switched while in state *Operation enabled*.

> Attention:
>
> Ensure that all required PDO data is available for the mode you are switching to.

Object ID	Name	Description
0x6060	Op Mode	Use this object to select the desired Mode

Extended Control Functionalities

Synapticon provides several features for enhancing motion control even with low-cost hardware:

Cogging Torque Compensation Removes ripples caused by magnetic interaction between motor rotor and stator.

Field Weakening The velocity range of electric motors can be extended by weakening the magnetic field of the motor linearly over the speed range.

Anti-Windup Control An Anti-Windup algorithm is used to prevent the Integrator part to accumulate beyond acceptable boundaries.

Gain Scheduling Applies controller gains which will change dynamically regarding the current speed.

Motor Overload Protection (i2t) This feature limits the overcurrent injected to the motor to prevent damage from overheating.

Velocity feed forward To increase the velocity of the robot's reaction, the target position signal can bypass the position controller.

Torque Offset A desired offset torque can be added to the reference torque.

Velocity Offset An additional velocity offset can be used to improve the positioning performance.

Filtering Digital filters can be used to remove unwanted frequencies such as noise or nonlinear effects.

Position and Velocity Feedback Low-pass filters

Torque loop input shaping filter

Command smoothing and Interpolation: Successive setpoints can be smoothed, so that low command resolution and jitter have reduced effect on the control.

Control supervision Allows to supervise the work of the control loops

Feed constant

Store home position

Distributed clocks Allows to synchronize the system time for enhanced performance.

OS Command Offers commands that can't be executed by single SDO operations

Quick stop Stops the motor quickly in a controlled way

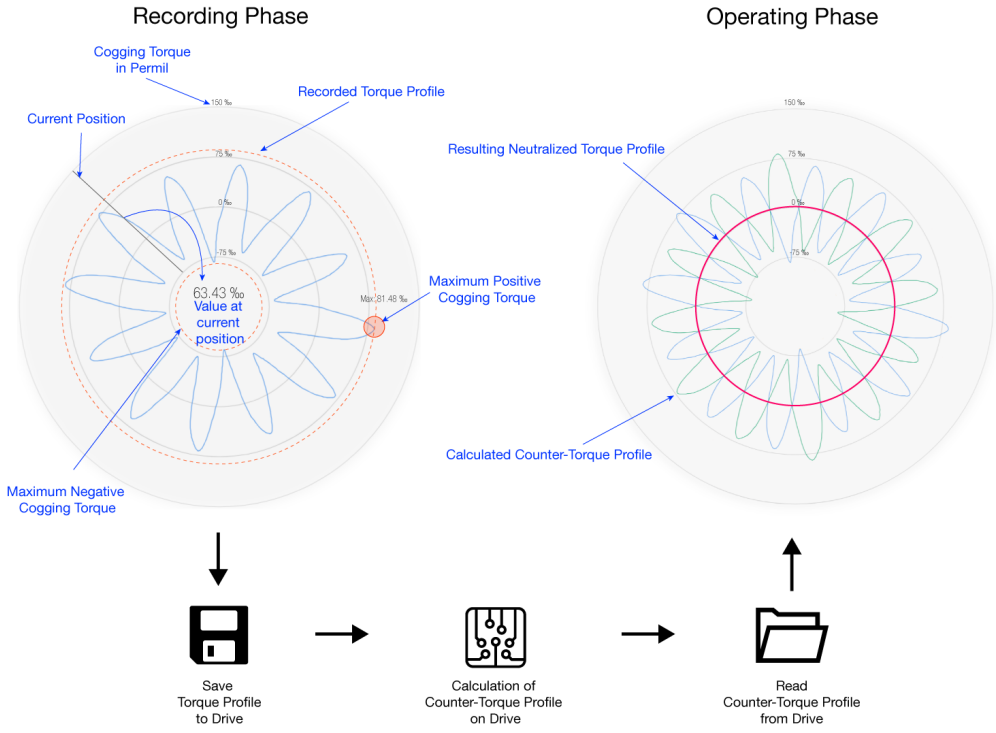
Cogging Torque Compensation

Overview

Cogging is an effect caused by magnetic interaction between the motor rotor and the stator. It results in torque ripples over the motor revolution, which leads to uneven motion or generally unsatisfying performance of the motor at low velocities. This phenomenon is of particular concern in low-cost motor models but can be compensated on a software level.

Cogging torque compensation is a software feature that suppresses torque ripple of electric motors to overcome vibration problems in high precision and low velocity applications. It can be found in OBLAC Box.

When cogging torque compensation is activated, an additional torque component is generated actively by the motor in order to compensate for the cogging torque. The compensation feature makes use of the fact that cogging torque is dependent on the single-turn motor position. To find the amount of compensation torque needed at each motor position, a recording procedure has to be performed before the compensation itself can be used.



Attention:

During the recording phase, the motor will do several turns at low velocities. Since cogging

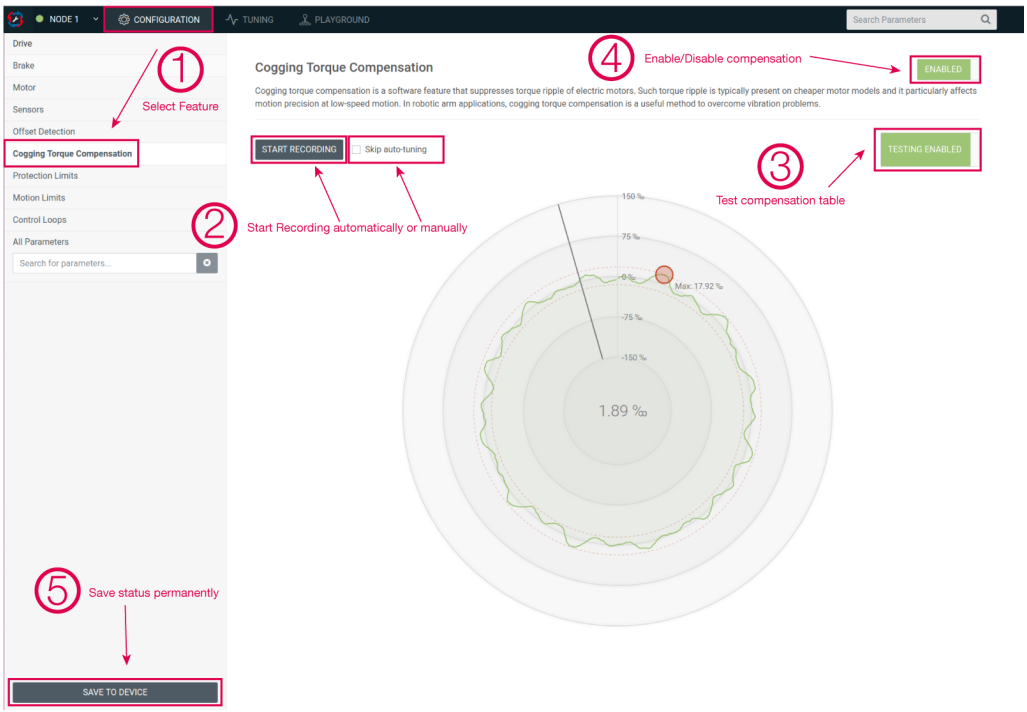
is a property of the motor and independent of attached loads it's possible to perform the procedure outside the actual setup.

Procedure

Prerequisites

- The minimum resolution of the encoder required is 1024 counts per revolution (cpr), it is recommended to use a resolution of 4096 cpr or higher.
- The motor and the drive should be properly sized. In particular, the motor's rated current should be at least 20 % - 30 % of the drive's rated current.
- For executing the cogging torque recording the motor must not be mechanically blocked and ideally must have no load attached.
- The motor is fully functional and the drive is properly configured.
- The System identification and position auto-tuning procedures were successfully executed with the current setup. The position control mode is fully functional.

Overview of the GUI



Recording

Before recording the cogging torque compensation table, a special configuration of the drive is required. This is done automatically, but can also be executed manually. When the recording is finished, this configuration is discarded and the old values are restored.

Automatic recording

This is the recommended default option that will do the procedure automatically. It uses a mathematical model of the electromechanical system obtained during the system identification to tune the controller. This procedure might take up to 3 minutes.

After setting up the drive and successfully executing the system identification, enter the menu item “Configuration” and select “Cogging Torque Compensation”.

To use the automatic recording, simply click on “Start Recording”

> Attention:
>
> During the recording routine the motor will do several turns.

Testing the compensation table

Once the recording is finished, the values are stored on the Drive. In the OBLAC Box GUI the feature is disabled and can now be tested manually:

1. Slide the switch to TESTING ENABLED
2. Turn the shaft manually, you will feel the torque ripples.
3. Set the cogging torque compensation feature to ENABLED
4. Turn the shaft again manually and feel the difference

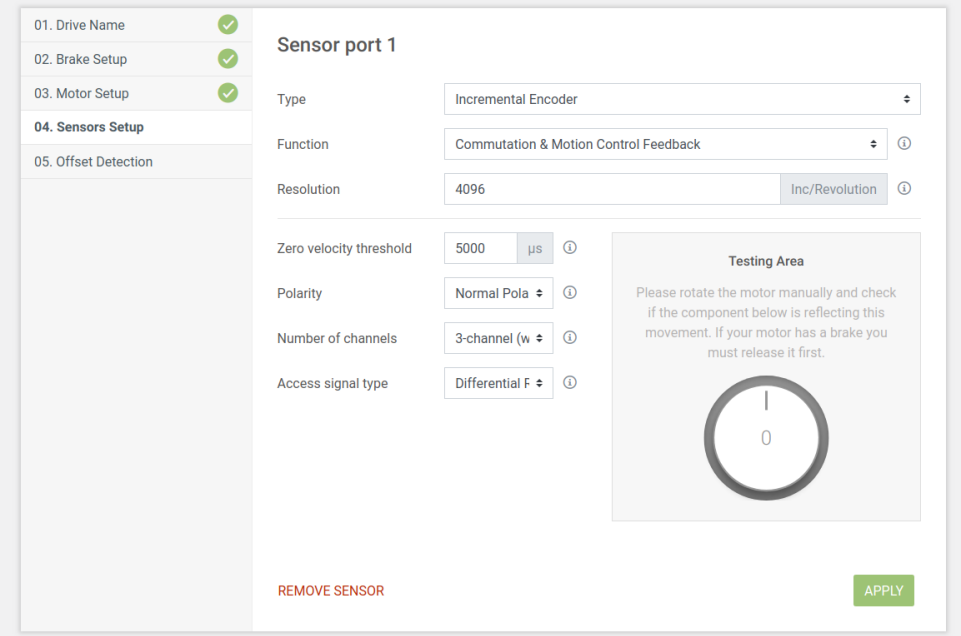
Manual configuration for the cogging recording

This option should be used when the performance of the auto-tuning is not satisfactory. In this case, the user should manually configure and tune the system. When the cogging torque compensation recording is triggered, the system will just use the current configuration.

Manually preparing the system for the cogging torque compensation recording:

1. An encoder on the motor shaft has to be used for the cogging torque compensation table recording. Set the function of this encoder to: "Torque", "Velocity" and "Position". This is

done in the setup wizard.



2. Go to the tuning section and perform System identification.
3. Select “Position control mode” and tune the cascaded control loop with the sliders.
4. The tuning should be as sharp as possible: try setting the settling time slider to 50-80 ms.
5. Verify the tuning by setting the position ramp profile to 10 RPM velocity. The actual velocity should be as close to 10 RPM as possible.
6. Go to the cogging torque compensation screen and check the ‘Skip auto-tuning’ checkbox.
7. After the recording is finished, test the obtained compensation table:

   Use switch ‘TESTING DISABLED’ in the cogging torque compensation screen to enable the torque control. Switch ‘DISABLED/ENABLED’ to enable/disable the cogging torque compensation table and feel the difference by rotating the motor shaft by hand.

8. Restore original configuration of the drive and re-tune control loops if possible.

Using the compensation table

Once the recording procedure is successfully finished, the cogging torque compensation can be activated and deactivated both in the OBLAC Drives GUI as well as via EtherCAT. The Object 0x2008 ”Cogging torque compensation is set when the cogging torque compensation is enabled or disabled in the GUI of OBLAC Box .

Set the switch for the cogging torque compensation to “Enabled” and click “Save to Device” to use the feature during operation of the drive.

The feature is activated globally and will be used in all modes of operation (torque control, velocity control and position control).

Controlling the feature via EtherCAT

To enabled or disabled the compensation algorithm Object 0x2008 ”Cogging torque compensation”, subindex 2 (cogging torque compensation enabled) must be set to 1.

By enabling the cogging torque compensation algorithm, the table communicated to the torque controller is added as a set of values to the torque reference value depending on the motor position. If the algorithm is disabled, this is skipped.

To start the cogging torque compensation recording procedure via EtherCAT, the manufacturer specific operational mode “COGGING COMPENSATION RECORDING” must be enabled in Object 0x6060 (Op Mode) by setting the value to -1.

Subitems:

1 State

State of the cogging torque compensation table. Possible values:

0	NO DATA	there is no cogging table in the memory
1	IN PROGRESS	
2	SAVING	recording is done and table is being saved to the memory
3	DATA EXISTS	cogging table exists and is saved in the memory

2 Enabled

Enables or disables the cogging torque compensation feature. Set to 0 to disable or to 1 to enable the feature.

The status of the Cogging torque compensation record (0x2008:1) must be set to 1 (IN PROGRESS).

During the start-up of the device it is checked if the cogging table exists in the flash memory and if it exists, the table is communicated to a torque controller. The state is generated accordingly:

- 0 (no table exists)
- 5 (table exists)

Parameters

Object ID	Name	Description
0x2008	Cogging Torque Compensation	Parameters needed for Cogging Torque Compensation

Field Weakening

In many applications, it is required to extend the velocity range of an electric motor. In-wheel Automotive applications, gearless wind turbines and fast robotics are good examples of such applications.

As shown in Fig. 1, “Field Weakening” is a feature that increases the speed range of a PM motor by weakening the magnetic field of the rotor linearly over the speed range.

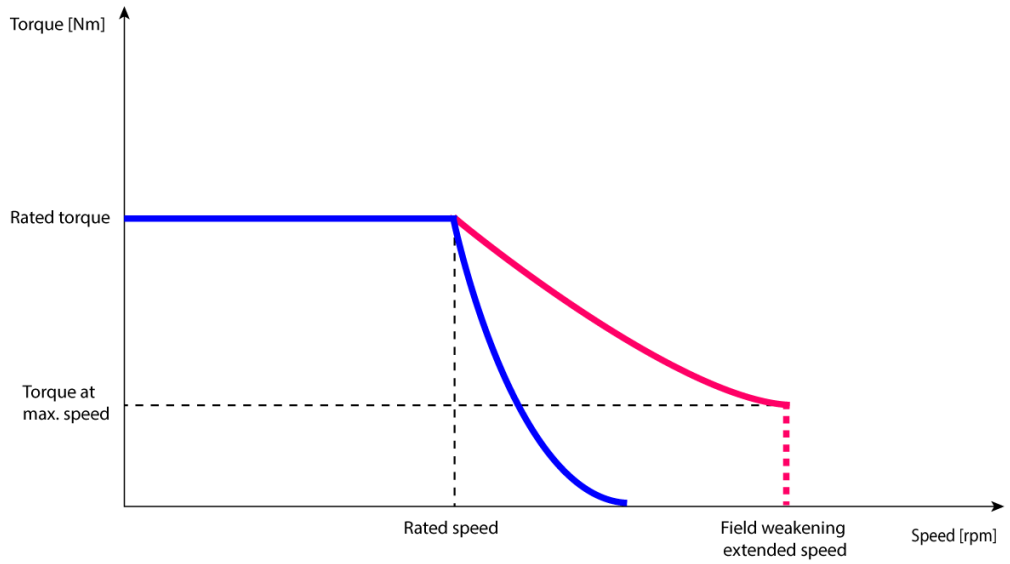


Fig. 1: Rated and extended speed range of a BLDC

Attention:

Use this feature at your own risk, it can seriously damage your hardware (including catching fire) when the values are set wrongly! Even though not all risks can be eliminate, these measures can reduce them:

- This is an expert feature! Please make sure you have sufficient knowledge about BLDC motors.

- Change all values gradually and with caution, check results often.
- Never use field weakening to achieve velocities that correspond to more than 80 Volts of peak back EMF voltage. Otherwise the drive may start burning!
- Avoid switching off the drive while in motion - especially at high velocities! Otherwise the drive may start burning!
- Avoid using the torque control mode when field weakening is activated. It is better to use velocity control or position control. Otherwise the system can become unstable, the drive may start burning or mechanical components may brake.
  Generally avoid using the torque control mode when field weakening is activated. It is better to use velocity control or position control. Otherwise the system can become unstable, the drive may start burning or mechanical components may brake.
  In some development or test scenarios, it is acceptable to still use the torque control mode, as long as there is a load machine connected to your motor or any other mechanism that will ensure that the speed cannot exceed the maximum allowable speed mentioned in (3)

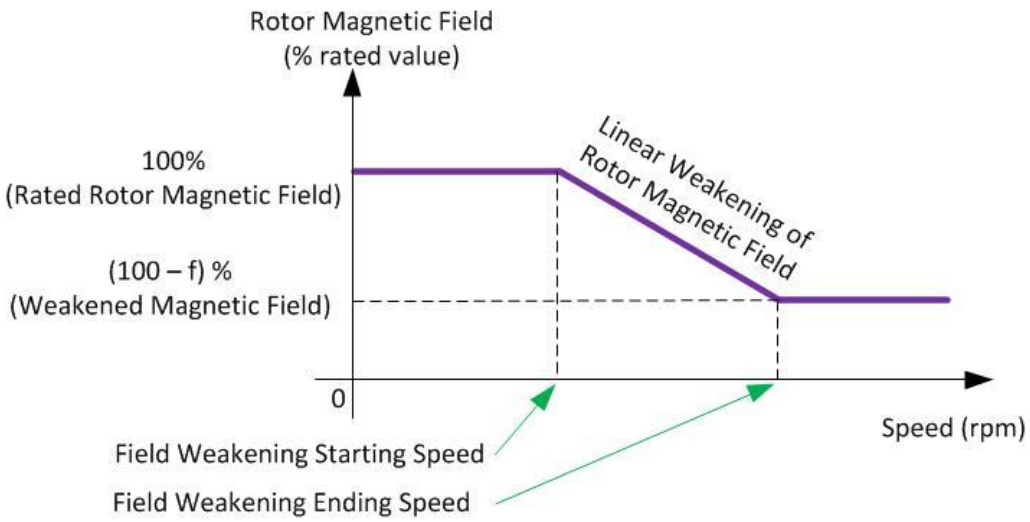


Fig. 2: Percentage of field weakening in respect to motor speed

As shown in Fig. 2, the user should set 3 parameters for this feature:

1. Field Weakening Starting Speed
2. Field Weakening Ending Speed
3. Percentage of Field Weakening (f)

Once the motor speed hits “Field Weakening Starting Speed”, the field weakening procedure starts to linearly reduce the magnetic field of rotor until the motor speed hits “Field Weakening

Ending Speed". The user can also determine how many percentage of rotor magnetic field should be finally reduced. As a result, the slope of rotor field reduction would be:

$$\frac{-f[\%]}{Field\ Weakening\ Ending\ Speed - Field\ Weakening\ Starting\ Speed}$$

> Note:
>
> The field weakening feature can not be enabled when the drive is in fault state.

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Parameters needed for Field Weakening can be found in Subindices 4-8 of the object Torque Controller

Anti-Windup Control

When the controller sends signals to the motor to accelerate at a very high rate, it's possible that the values are beyond the motor's capabilities. This is likely to happen in the case of large and long-lasting control deviations. If the controller has no knowledge about the motor's physical properties, the signal would remain saturated, even after the control error has been compensated: the motor would still accelerate at maximum power, which would result in large overshoot or even instability. Saturation limits for the integrator are introduced and can be set in the controller options (Integrator boundaries) to limit the integrator output.

The Anti-windup control algorithm is implemented with the following approach:

$$u_i(k) = u_i(k-1) + K_I \cdot T \cdot e(k)$$

$$if\ |(u_i(k))| \geq l,\ then\ u_i(k) = u_i(k) - K_I \cdot T \cdot e(k)$$

with:

e: control deviation

u_i: Output of the integral part

T: Sample time

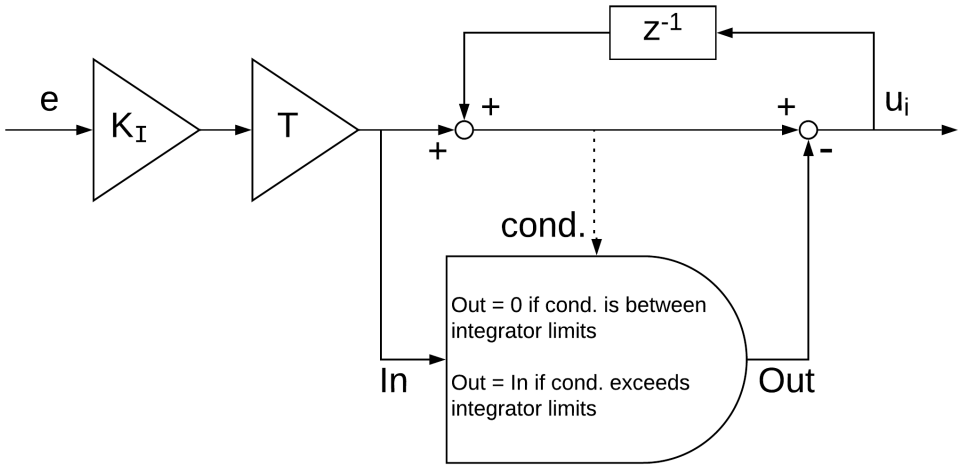
K_i: Integral gain

I: Integrator limit

Explanation: If the integral part of the controller goes higher than the value of the controller's output limit, the value that was previously added to integral is automatically subtracted.

This method is known as "clamping". It requires no tuning.

Detail view on the **integrator branch** with anti-windup:



The described implementation of the integration algorithm is used in all motion controllers (both integrators of the cascaded position controller, integrator of the PID position controller, integrator of the PID velocity controller). The variable *Integrator limit* is defined by the subitems 2012:4, 2012:8 and 2011:4 respectively.

Parameters

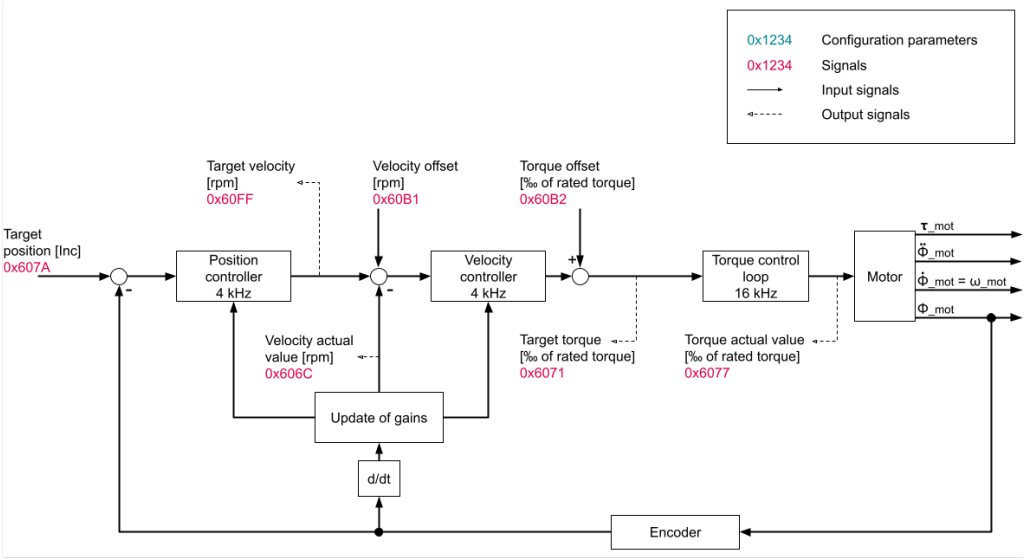
Object ID	Name	Description
0x2011	Velocity Controller	Values K_p, K_i, K_d and Integrator Limit for Velocity Controller
0x2012	Position Controller	Values K_p, K_i, K_d and Integrator Limit for Cascaded Position Controller (Both loops: Position and Velocity)

Gain Scheduling

Depending on the application, the "optimal" tuning for the cascaded PID controller may differ between different operating points. If there is only one tuning available, it has to be a compromise

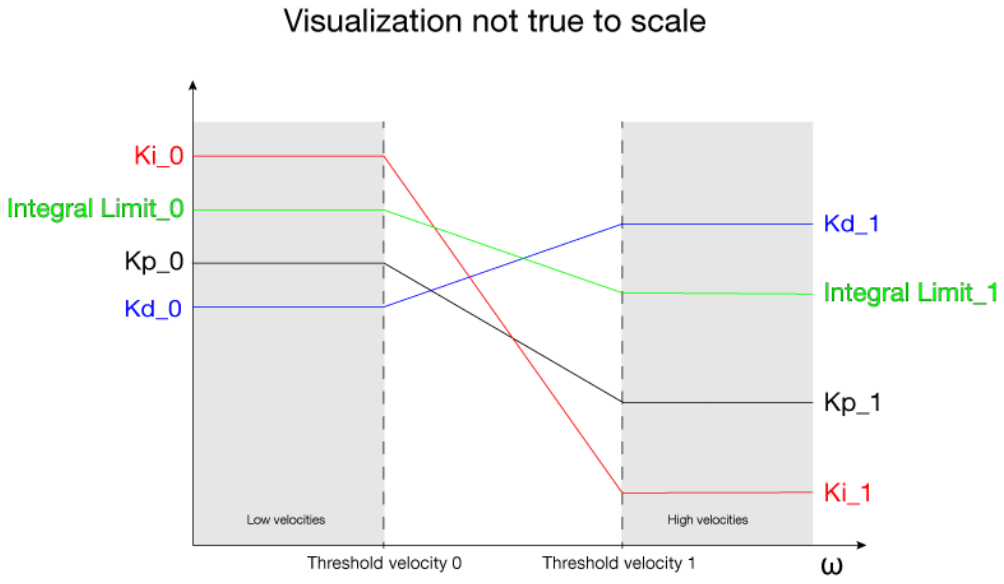
not using the full potential of the hardware.

This feature allows the user to apply controller gains which will change dynamically regarding the current speed. This technique can be used within the cascaded position controller to compensate for nonlinear effects, especially stiction / friction in the mechanics.



For velocities (in rpm) below a user-defined value, a set of PID values determined best for slow motion is used. Likewise, for velocities (in rpm) above a user-defined value, a set of PID values determined best for fast motion is used.

The PID values between these values are **interpolated linearly.**



> Note:
> Please note that this feature is at the moment experimental. It is currently accessible only directly via EtherCAT through the parameter 0x2013:1 … 0x2013:18. The gain scheduling feature is only available in Cascaded PID mode which is selected in object 0x2002.

Parameters

Object ID	Name	Description
0x2002	Position control strategy	Selects the control structure used for position control
0x2013	Gain Scheduling	Parameters needed for Gain Scheduling

Motor Overload Protection (i2t)

Overview

This feature will evaluate the “overheating” status of the motor. If the motor is overheated more than the allowed limit, this feature will limit the RMS value of phase currents to protect the motor from being overheated beyond the permitted limits.

> Note:
> This feature doesn’t directly measure the motor temperature. It is only a limiting feature which doesn’t allow the motor to overheat.
> It is able to detect fast transients and can therefore react much faster than the attached temperature sensors.

Requirements

To use this feature, the following parameters must be set properly according to the motor data sheet:

Object ID	Name	Description
0x6075	Motor rated current	Motor rated current (rms value)
0x6073	Motor maximum current	Motor maximum current (rms value)
0x200A	i2t	Parameters to activate the i2t feature

Description of Subitems for 0x200A:

Object ID	Name	Description
0x200A:1	Enable	Set to 1 to enable the protection
0x200A:2	Peak Time	Maximum time period [ms] in which the motor is allowed to have its maximum current
0x200A:3	Percent max	The percent of the maximum transient energy allowable in the motor
0x200A:4	Protection active	When this flag is set, the output current is limited to equal or less the nominal current

Operating principle

The difference between copper losses under rated conditions and overload conditions is calculated, and is used to limit phase currents in case of motor overheating.

Once the motor phase currents are higher than their rated values, the overheat calculation starts by integrating the difference between losses under rated and overload conditions.

When a current is injected that is above the nominal current, the tracking variable will rise. If the overcurrent is less then the maximal value the peak time is extended accordingly.

Once the tracking variable reaches 100%, the calculated thermal energy storage is reached and the output current in all phases will be limited to their rated values.

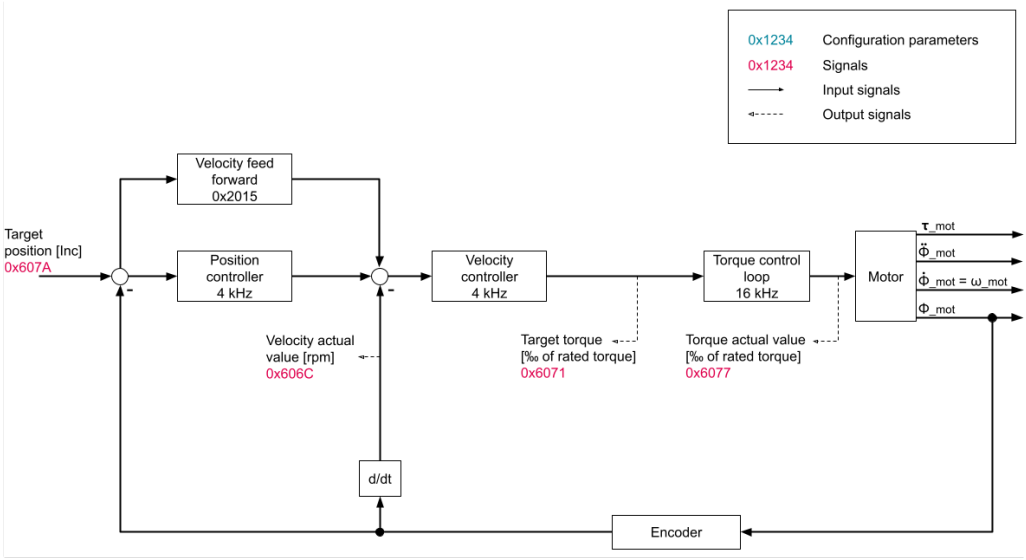
When the output current has fallen below the continuous limit, the difference between the actual current and the continuous limit current will be negative and the i2t tracking variable will decrease.

When the i2t tracking variable falls to 50%, current limiting is turned off and the motor phase currents can reach higher values again.

> Attention:
>
> The tracking variable in subitem 0x200A:3 has a maximum value of 250%. Values that exceed this limit are ignored. If the drive is used at values above 250%, the thermal energy in the motor is completely untracked!

Velocity feed forward

Velocity feed forward allows to increase performance as the target position signal gets routed directly to the velocity controller so that the velocity signal can be generated faster. After differentiation, the signal is filtered by a first-order low-pass filter and multiplied by a configured gain.



> Note:
>
> The filter is used to smooth the derivative of the target position. This is needed to soften communication lags (if present), attenuate noise in the target position signal and diminish the influence of target position quantization.

Usage

- Default value of gain is 0 (This means Velocity feed forward is off)
- The gain is given in per mil 1. A value of 1000 is a good starting point.
- Reduce the cutoff frequency if periodic torque spikes are observed during motion. The default value is 1000 Hz.
- Adjust gain and filter cut-off frequencies until torque distortions during motion are acceptable.
- To switch off the filter enter a very high cut-off frequency.

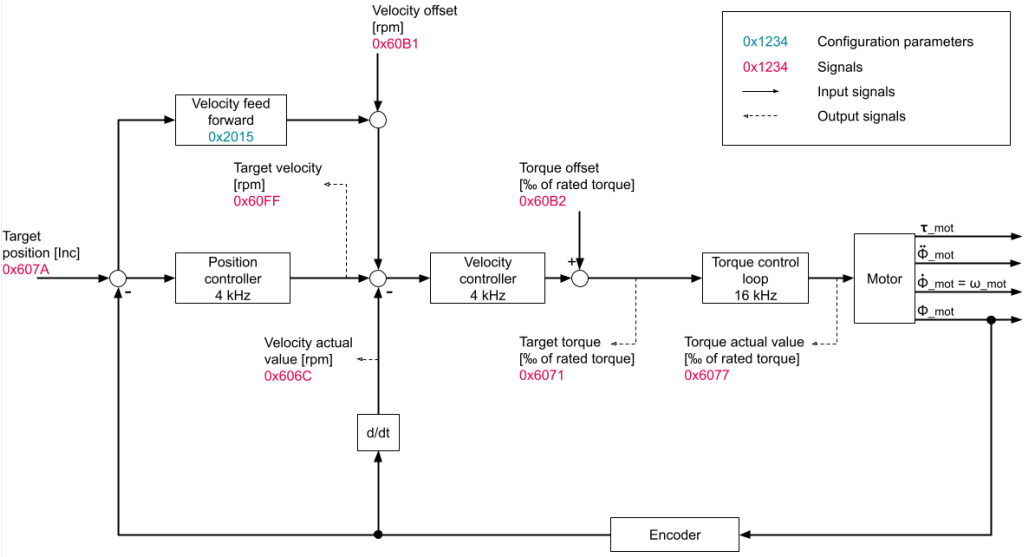
1

1000 means the target position signal will be routed directly, 500 means scaled to half

> Note:
> It is highly recommended to use an encoder with a resolution of more than 14 bit per revolution. Using encoders with a lower resolution will result in computed velocity quantization and torque spikes during the acceleration of the motor. Using a dual loop control might result in a similar effect if the gear reduction ratio is high and the resolution of the driving shaft encoder is small. In these cases, the feature still can be used with a very low cut-off frequency (5-10 Hz). Alternatively, a master-generated feed forward velocity can be sent to the slave via the velocity offset object.

Using a velocity offset additionally

Additionally, a velocity offset can be added to the control loop for even faster movements with high dynamic.



Click here for details for using the velocity offset.

Object used for Velocity feed forward

Object ID	Name	Description
0x2015	Velocity feed forward	In this object the cut-off frequency of the low-pass filter and the gain are configured

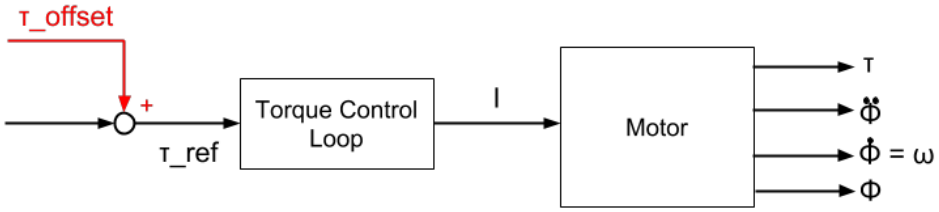
Torque Offset

This feature can be used to improve the performance of highly dynamic motion.

> Use case:
>
> Dropping a load
>
> If a load torque in a joint suddenly decreases, an involuntary movement can occur when the control algorithm tries to compensate. To avoid this peak in torque deviation, the torque-offset object can be used. The central application controller “knows” when the robot will drop the payload and how much this payload is. The robot master controller can use the kinematic model of the robot to compute how much torque this payload will add to each of the joints. So it can command these torques as offset torque to each joint. So right at the moment when the robot drops the payload, it will exert the exact amount of compensation

orque that is required to stay in its desired pose.

On top of the desired velocity input, the position and velocity controllers also accept a desired ‘off-set torque’, which is a dynamically changeable torque value that will be added to the torque value generated by the velocity PI controller. This applies to the simple PID position controller, to the cascaded position controller and to the velocity controller. The offset torque can be updated at full EtherCAT communication frequency (4 kHz in most configurations) and is given in the unit of per thousands of rated torque.



To use this feature, the high level robot control must contain a model or any other knowledge about the system’s physics, e.g. acceleration feed forward or gravity compensation.

The robot master controller can use the kinematic model of the robot to compute how much torque this payload will add to each of the joints. So it can command these torques as offset torque to each joint. So right at the moment when the robot drops the payload, it will exert the exact amount of compensation torque that is required to stay in its desired pose.

Note:

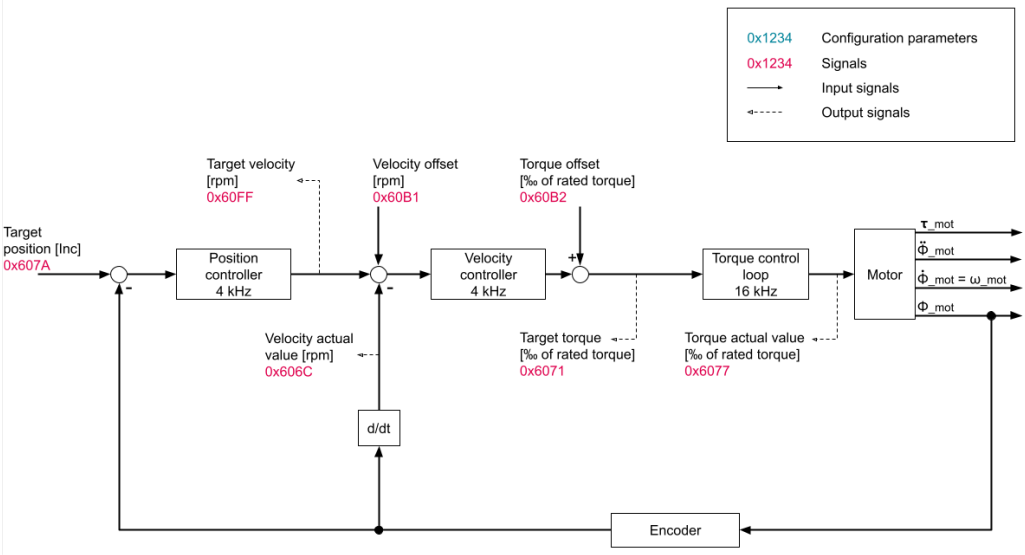
As this value should be fed dynamically with continuously updated values, it is not accessible in OBLAC Drives, but through the EtherCAT object 0x60B2.

Parameters

Object ID	Name	Description
0x60B2	Torque Offset	This object provides the offset for the torque value

Velocity Offset

To improve the positioning performance when the position reference is changing dynamically, an additional velocity offset is available in the cascaded controller.



This velocity offset must be computed by the master, it is not computed on the slave.

> Note:
>
> As this value should be fed dynamically with continuously updated values, it is not accessible in OBLAC Drives, but through the EtherCAT parameter 0x60B1.

Parameters

Object ID	Name	Description
0x2010	Torque Controller	Values K_p, K_i and K_d for Torque Controller
0x2012	Position Controller	Values K_p, K_i and K_d for Cascaded Position Controller (Both loops: Position and Velocity)
0x60B1	Velocity Offset	This object provides the offset for the velocity value

Filtering

Overview

> Attention:
>
> Since Firmware 5.0 only 1st order Position and Velocity Feedback filters are supported.

A digital filter removes unwanted components from the signal which passes through it.

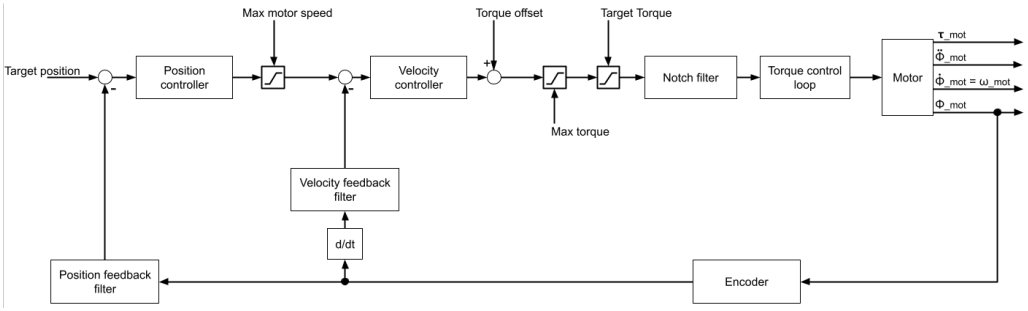
This feature can be used for example as:

- Position Feedback filter - Reduce encoder noise. A 1st order low-pass filter is supported.
- Velocity feedback filter - Reduce noise and effects of encoder nonlinearity in the velocity signal. A 1st order low-pass filter is supported.
- Loop shaping filter - Suppress excitation of resonance frequencies in the output of the motion control loop or in the input of the torque control loop. A 2nd order notch filter is supported.

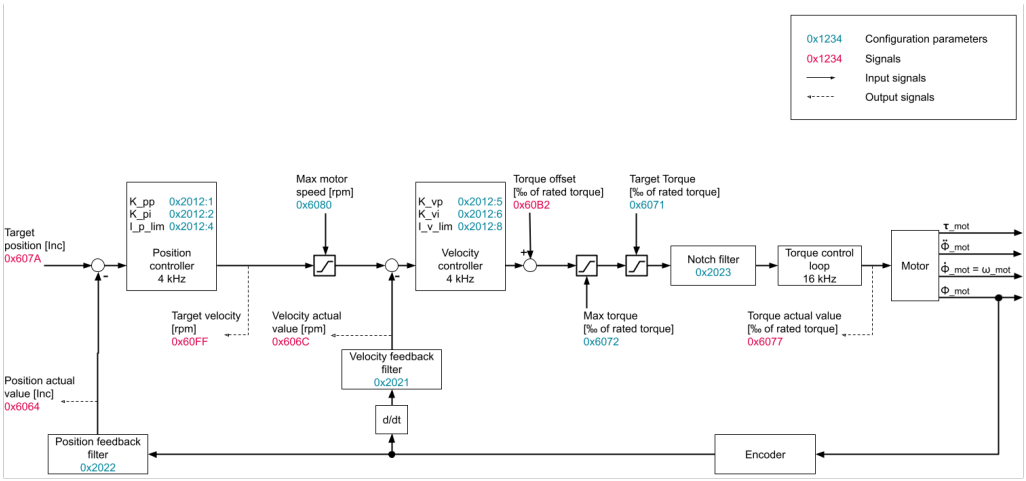
Any signal can be decomposed into a sum of sine waves with different frequencies. Some of them might negatively affect the control loops. It is possible to attenuate these unwanted frequencies by properly configuring a corresponding filter.

Configurable filters are implemented for the signals that are considered the most important for motion control.

General View



Detail View



The position feedback filter can also be used with a simple PID loop:

Methods

Definitions

- s - Laplace-domain frequency parameter
- f_c - Cutoff frequency of the low-pass filter or central frequency of the Notch filter, defined in Radians/s [1]
- f_w - Full width of the rejection band of the Notch filter $\frac{f_c - f_w}{2}, \frac{f_c + f_w}{2},$

  defined in Radians/s [0]
- ζ

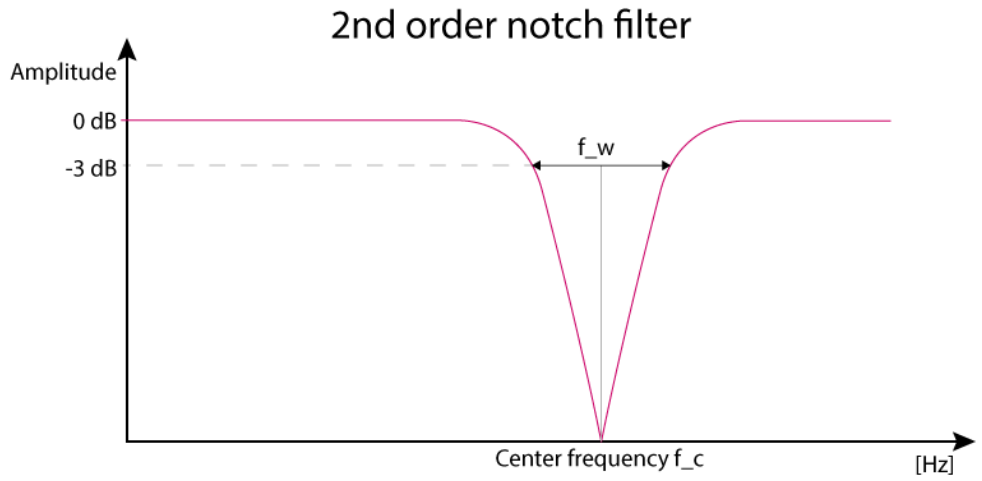
  - Damping ratio

  [0] Radians are used for internal calculation, user input is in [Hz]

[1] My new footnote
[0] My new footnote
[0] My new footnote

Low-pass filter

Torque loop input shaping filter



The following equation was used to design the torque loop input shaping filter:

$$BS2(s) = \frac{s^2+f_c^2}{s+f_w s+f_c^2}$$

Application Guide

Position and Velocity Feedback Low-pass filters

Torque loop input shaping filter

Parameters

Object ID	Name	Description
0x2021	Velocity feedback filter	Cut-off frequency of velocity feedback filter
0x2022	Position feedback filter	Cut-off frequency of position feedback filter
0x2023	Notch filter	Center frequency and bandwidth of notch filter

Position and velocity feedback low-pass filters

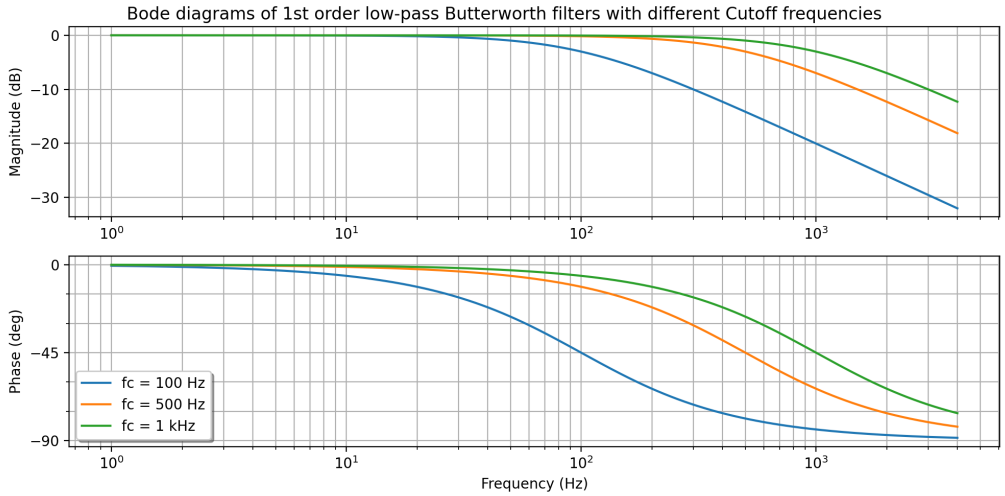
These filters are used to cope with distorted feedback signals that occur in the control loop. Typical distortions to consider are:

- Encoder noise in the position signal - optical and good magnetic encoders usually have a low amplitude. 1st order filters with a 1000 Hz [1] cutoff are a good starting point.
- Noise in the velocity signal - noise from the position signal is amplified due to differentiation. 1st order filters with a 500 Hz * cutoff are a good starting point.
- Velocity signal distortions due to encoder nonlinearity. These distortions typically appear with a frequency of 1 (dominant), 2 and 4 * motor speed [revolutions per seconds].

[1] The specified frequencies are just initial values that will not interrupt the motion control. Lower values should be found and set; see A practical guide to low-pass filter design.

> Note:
>
> Setting the velocity filter more rejective than the position filter is one possible approach, see A practical guide to low-pass filter design for details how to set up filter parameters.

The main parameter of the Low-pass filters is the cutoff frequency which determines the frequency where the filter starts to attenuate the signal to -3 dB of the amplitude. After that frequency, a roll-off region starts that further attenuates the signal.



> Note:
>
> Filtering comes at a cost of a phase shift. This might significantly limit the maximum sharpness of the control loop.

A practical guide to low-pass filter design

1. Determine the approximately required bandwidth of the control loop. This can be done based on the reference profile specifications or roughly estimated from the actual rise time during a step response test:

   $$Bw \simeq \frac{0.35}{t_r}[Hz]$$

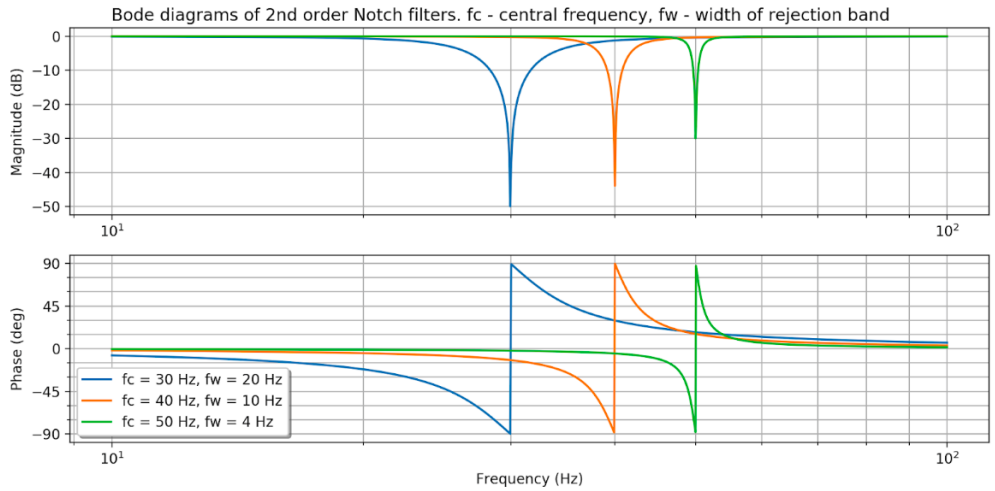
2. Ideally, the filtered signal shouldn't be modified until this frequency. The magnitude change and phase shift caused by the filter have to be as little as possible. Thus, the filter cutoff frequency should be approximately one decade higher (Bw * 10).
3. Examine the spectral density plot of the target signal and determine the frequency range that has to be filtered out.
4. Enable a filter with the default cutoff frequency.
5. Try to change the cutoff frequency until an appropriate balance between noise filtering and phase-shift/magnitude attenuation is reached.

Parameters

Object ID	Name	Description
0x2021	Velocity feedback filter	Cut-off frequency of velocity feedback filter
0x2022	Position feedback filter	Cut-off frequency of position feedback filter

Torque loop input shaping filter

If the mechanical system has a resonance (for example a natural frequency) or if the system contains another source of distortion at a known frequency, exciting this particular frequency with the output of the torque controller may result in unwanted oscillations. This filter can be used to remove a specified frequency band around the resonance.



In practice, there is a certain challenge in usage of notch filters. If the resonance frequency is too low, introducing a notch filter will impair the control signal by adding a **phase delay**. It is therefore recommended to **use notch filters only for high frequency resonances**. *

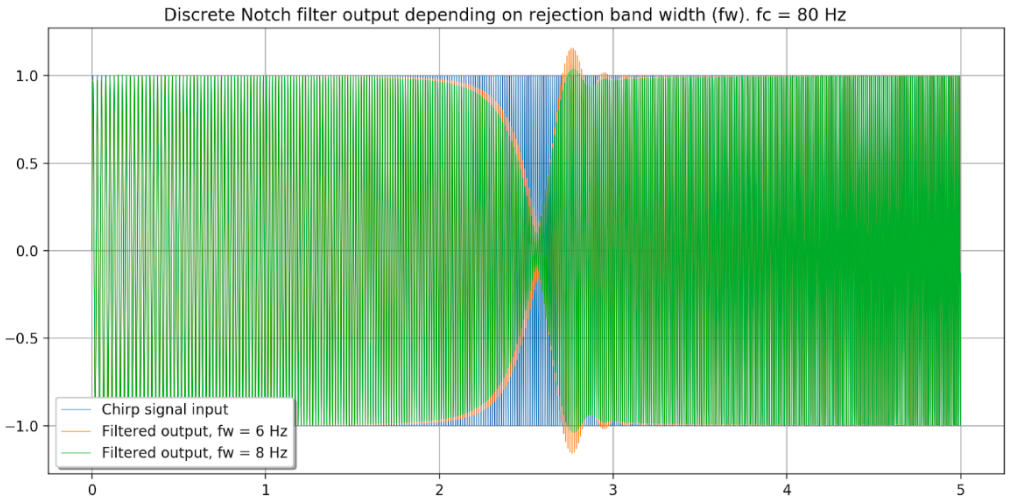
* The frequency depends on both the central frequency and the rejection width. Generally, it should be higher than 50 Hz.

A practical guide to notch filter design:

1. Find resonances of the system. This can be done by observing a frequency response during the system identification procedure: Check the velocity reaction to a specific torque input frequency.
2. Decide which resonance should be compensated and identify the width of excitation frequencies.
3. Record the reference behavior of the system before applying the notch filter.
4. Configure the filter with the obtained parameters.
5. Check the resulting performance of the setup and optimize the filter parameters.

> Note:
>
> The **smaller the width** of the rejection band, the **more ripples** are in the surrounding frequencies amplitudes. However, the area of the phase shift is smaller as is the signal attenuation at the rejection band. It is recommended to **use a rejection band width of at least 8 Hz** in order to avoid overshoot.
>
> The bigger the width of the rejection band, the smoother is amplitude transition around the

central frequency. Additionally, the attenuation is bigger. This happens due to the second order of the filter



Parameters

Object ID	Name	Description
0x2023	Notch filter	Center frequency and bandwidth of notch filter

Command smoothing and Interpolation

Overview

Commands are sent to the slaves with a step size that depends on the master's update rate. In EtherCAT, these steps are usually in intervals of 250 µs to 8 ms while other protocols may have even slower rates.

These steps may cause audible noise or even instability when a controller is tightly tuned.

Fast command changes and slow update rates increase the negative effects on control performance.

To improve the quality of the trajectory, the input command can be shaped with a finite impulse response filter.

This filter smoothes the control values from the previous to the current target, smoothing out the input and preventing spikes in the derivative around the target changes.

This feature can be used in all position and velocity control modes:

- Cyclic synchronous position mode
- Cyclic synchronous velocity mode
- Profile position mode
- Profile velocity mode
- Homing mode

Side Effects

> Attention:
> This feature introduces delay to the control input therefore it's important to close the control loop in the servo drive directly. If the control is closed on the master, this feature would introduce a lag that may affect that controller's stiffness.

> Note:
> All axes on the chain must be commanded together and should have the same FIR settings.

Applications

Input interpolation

The master is configured to use the distributed clock so that all input commands are equally spaced in time. The FIR filter is set to operate such that the filter order is equal to the number of control cycles between two successive target command updates. This ensures the command is fully respected, since the last step of interpolation hits the requested point.

Pros

- The controller "ramps up" to the next commanded position instead of seeing a step function.
- The controller can be tuned much tighter than it would normally
- Removes noise that can occur when a tightly controlled system receives a step input.

Cons

- See Side_Effects

Jerk limiting

When the input trajectory contains undesirable discontinuous or aggressive motion, the performance of the mechanical system can be unsatisfactory.

Fixing the input trajectory can be difficult if the trajectory is built from sensor input and dynamically updated. Some trajectory generators are very basic, and smoothing the input is ideal to avoid poor motion.

The FIR filter should be set to smooth the input to the desired magnitude. Higher filter order will result in a smoother trajectory. Lower filter order will result in closer following of the input command.

Pros

- An easy way to avoid the need for complex trajectory generators on the master.
- Stable smoothing that guarantees settling on the input command by the end of the filter order.

Cons

- Smoothing will prevent the actuator from following the commanded trajectory, which may be problematic when part of a trajectory needs to be followed closely while other parts should be smoothed.
- End effector precision may be poorly affected for robots with serial kinematics.

> Note:
>
> The disadvantages above can be mitigated by slowing the trajectory in areas that need more precision.

Communication jitter mitigation

When no distributed clock is available, the clocks on the slave and master devices will drift, and this will cause a cycle to periodically either miss a command, or complete before the command arrives to the controller. This will effectively either double the input command or see no change to the input command. This creates a discontinuity that is twice the magnitude of the normal command steps.

Set the filter to an order similar to what should be chosen for input interpolation.

Pros

- While distributed clock and interpolated input will yield the best performance, using the FIR filter here will also smooth the discontinuities caused by communication jitter.
- The filter order can be boosted for additional smoothing if the communication delays are severe.

Cons

- See Side_Effects
- The trajectory is not guaranteed to match the input, since large steps will still cause “bumps” in the command. These bumps will be much less aggressive, but they won’t be entirely removed from the input.
- Extra smoothing will always yield extra deviation from the original command.

Detailed Description

A boxcar filter with rectangular gains is implemented, equivalent to a moving average filter (MAF). A moving average filter is equal to a linear interpolation if the filter order is equal to ratio between master and slave sample time - 1.

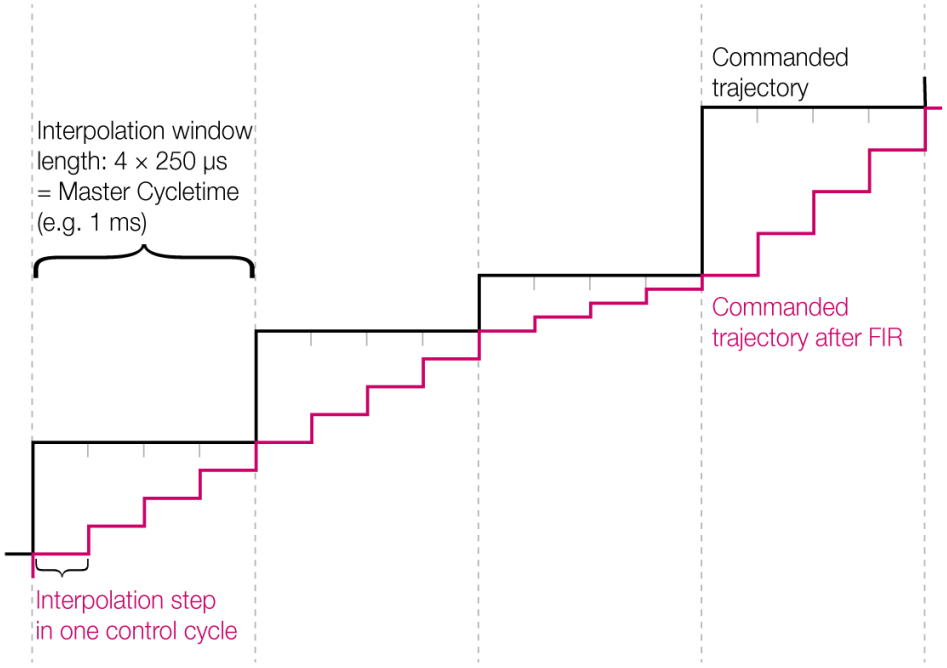
$$Filter\ order = \frac{master\ cycle\ time}{slave\ sample\ time} - 1$$

An example of the filter order for a master with 1 ms cycle time and 250 µs cycle time SOMANET slave:

$$3 = \frac{1ms}{250\mu s} - 1$$

The filter order is defined in multiples of 250 µs (e.g. 3 for a master with a 1 ms cycle), 3 is the default value that suits applications with 1 kHz EtherCAT masters perfectly. This number defines the length of the ring buffer that will be used as a MAF storage. At each 4 kHz cycle, the values in the buffer are summed up and divided by the buffer length to produce the output.

Command smoothing



Usage

In object 0x2027, the expected update rate and/or desired smoothing can be configured:

- Subitem 1 switches the feature on (1) or off (0)
- Subitem 2 selects the order of the filter

The Order is allowed between 0 (no action) and 31. The filter is enabled by default with an order of 3.

The order reflects the amount of steps the algorithm “looks back”.

Parameters

Object ID	Name	Description
0x2027	Control input FIR filter	Parameters needed for Finite impulse response Filter

Control supervision

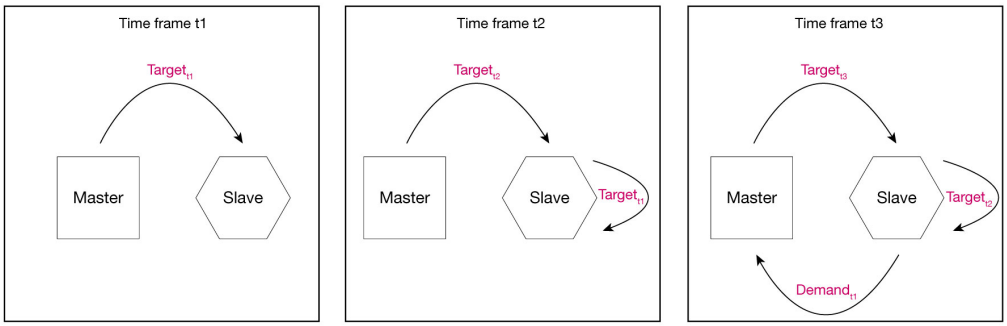
This feature allows to supervise the work of the control loops. This includes monitoring the internal variables and basic analysis of the control loop performance. Some monitoring functions and data objects are available only in specific control modes.

The main supervision functions are:

- Following error - Basic function that needs to be calculated on the master
- Following error window - Advanced feature that raises a flag in the Statusword
- Target reached function - Raises a flag in the Statusword

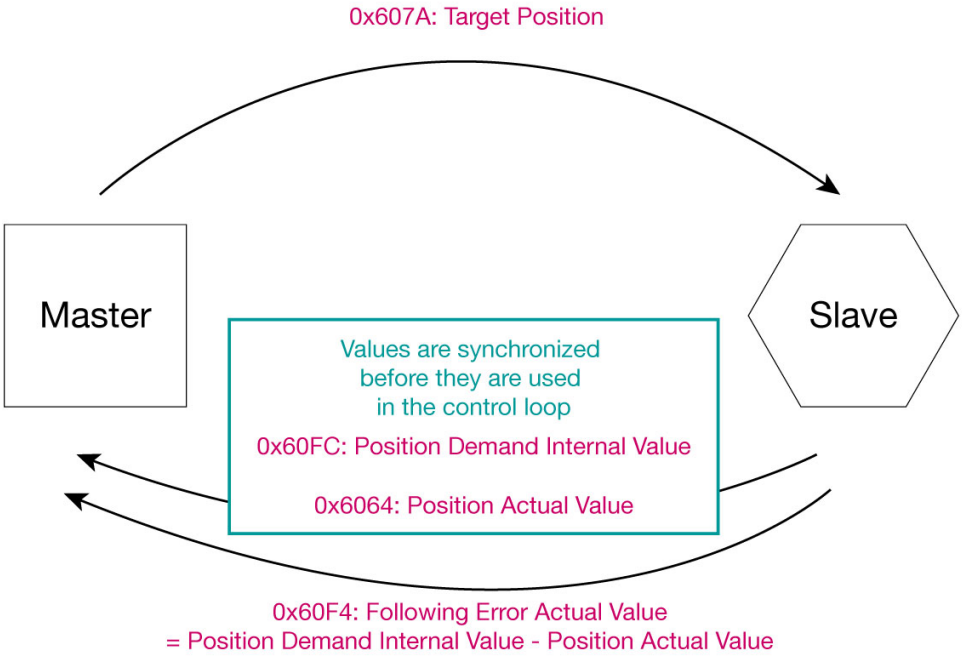
Timing diagram of the supervision objects

Two communication cycles after the master sends a target value to the slave (position/velocity/torque), the slave sends back two values: the Demand Value, indicating the value that was received by the slave and the Actual value as read from the sensors. These values are synchronized and sampled from inside the control loop.

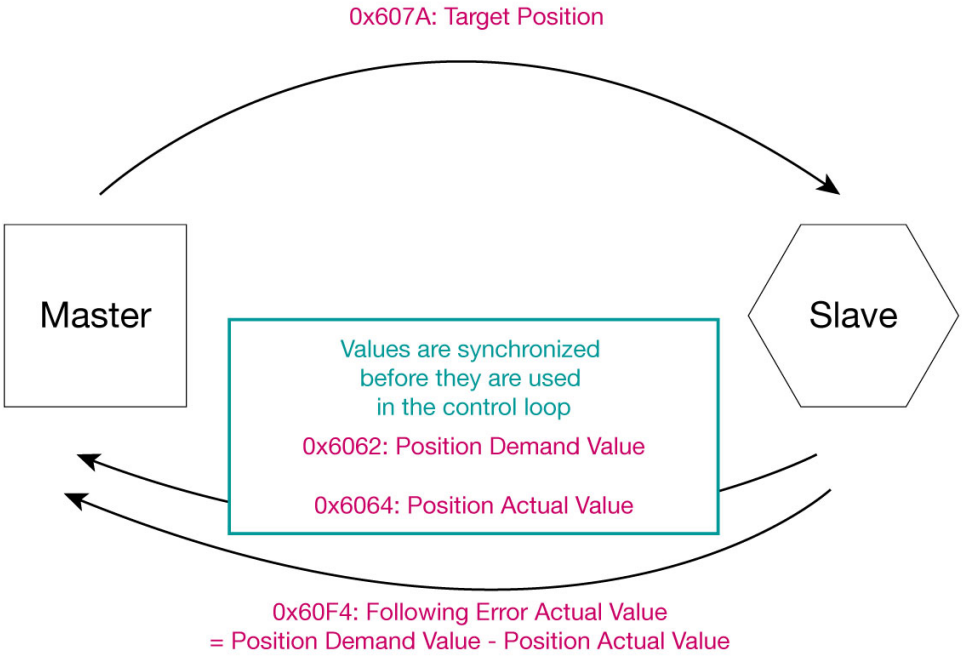


Name and designation of these objects differ, depending on the chosen mode of operation:

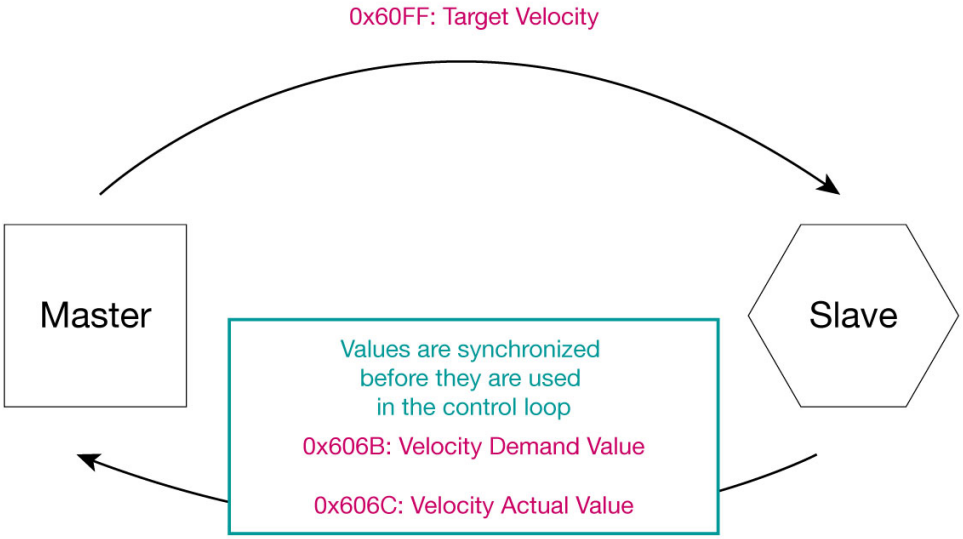
Profile Position



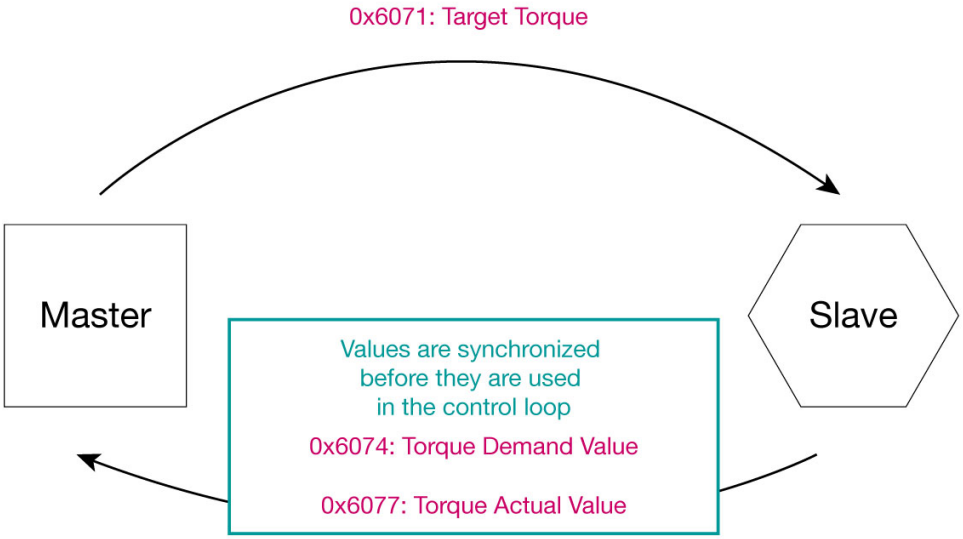
CSP


0x607A: Target Position
Master
Slave
Values are synchronized
before they are used
in the control loop
0x6062: Position Demand Value
0x6064: Position Actual Value
0x60F4: Following Error Actual Value
= Position Demand Value - Position Actual Value

CSV


0x60FF: Target Velocity
Master
Slave
Values are synchronized
before they are used
in the control loop
0x606B: Velocity Demand Value
0x606C: Velocity Actual Value

CST



Note:

In Profile Position mode, the master does not send a new calculated position at each cycle, instead, the slave computes the intermediate positions itself directly at the trajectory generator.

Therefore, the objects sent back from the slave to the master are different than in CSP.

Following error

The easiest way to measure the performance of a control loop is to calculate a following error:

Following error = Target value - Actual value

If the master-generated value is used as a target value, it is not synchronized with the actual measured value. The typical delay can be up to 3 communication cycles. Thus, the computed error will contain an extra error due to communication delay.

To circumvent this issue, ‘Demanded value’ objects should be used. These values a sampled synchronously with the actual measured value and should be used to correctly compute the following error.

Example: Velocity following error = Velocity demanded value - Velocity actual value

Alternatively, the object Following error actual value (0x60F4) can be used to monitor the position control loop. This object is also used for the Following Error Window feature.

Following error window

If the actual value is outside of the configured window for a configured period of time, the user is informed that the following error is bigger than the configured value.

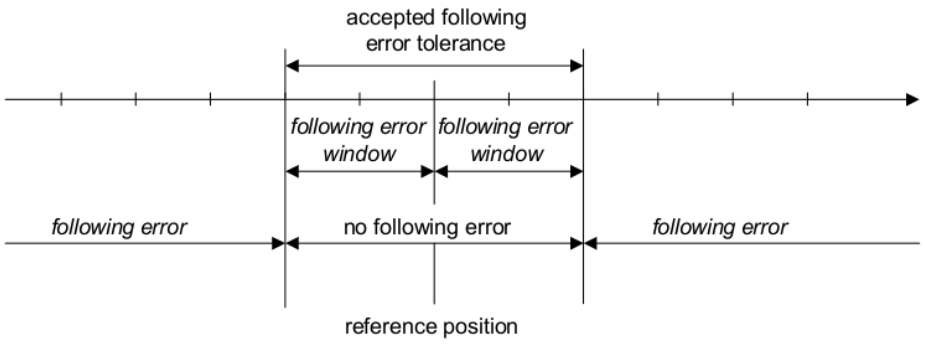
This information is reflected by bit 10 of the statusword.

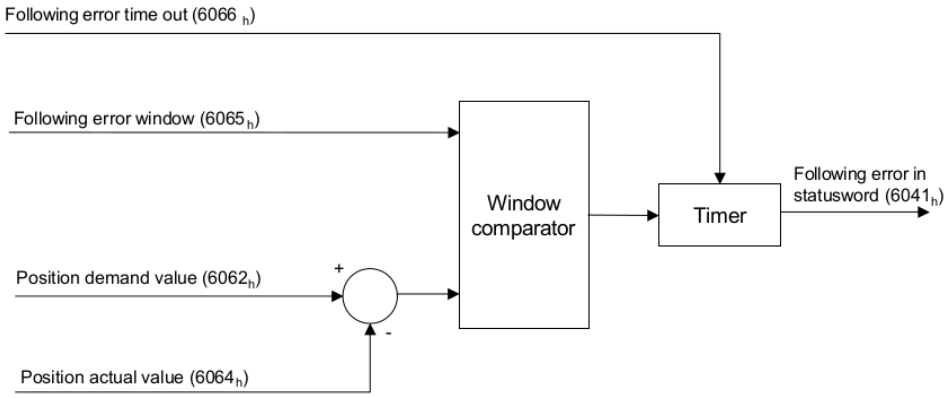
Different options are available for the subsequent action.

> Use case:
>
> Predictive maintenancing (monitor system wear)
>
> In case the following error grows bigger in time, check if mechanical wear is the reason.

> Note:
>
> The Following error window feature can only be used in a position control mode (operational modes CSP and PP)

Definitions





Configuring the Following error window

- Following error window 0x6065 [inc]: Use this object to indicate half of the configured range of tolerated position values symmetrically to the position demand value. If the position actual value is out of the following error window, a following error occurs. If the value of the following error window is FFFF FFFF h , the following control is switched off (default).
- Following error timeout 0x6066 [ms]: Use this object to indicate the minimum time that the actual position must be out of following error window so that the bit 13 of the statusword is raised to 1, indicating that a Following error has occurred.

Result

- Bit 13 of the Statusword indicates that the following error is outside the allowed window:
    - 0: value within error tolerance
    - 1: value exceeds error tolerance
- Following error actual value 0x60F4 [inc]: Provides the actual value of the following error.

Following error option codes

These codes are to configure the drives reaction in case of a following error that exceeds the window.

Value	Definition
0	Drive will report Following error in statusword (bit 13) and stay in operational state
1	Drive will report Following error in statusword (bit 13), perform a quick stop and enter a fault state

Value	Definition
2	Drive will report Following error in statusword (bit 13) and immediately enter a fault state

Target reached function

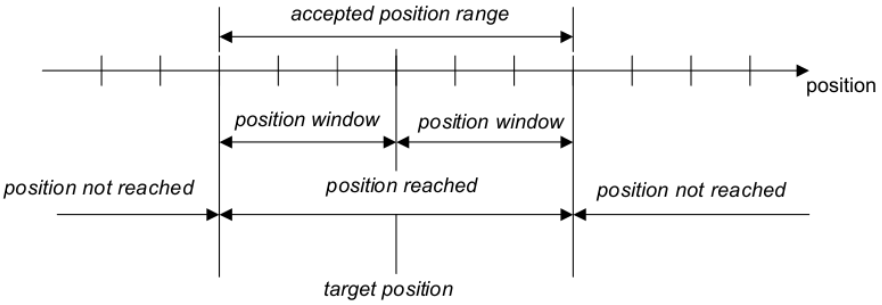
This feature can be used to inform the user if the requested target was reached (for example if the target position was set in the Profile mode). A specified target (position, velocity or torque) can be defined with a margin at each side.

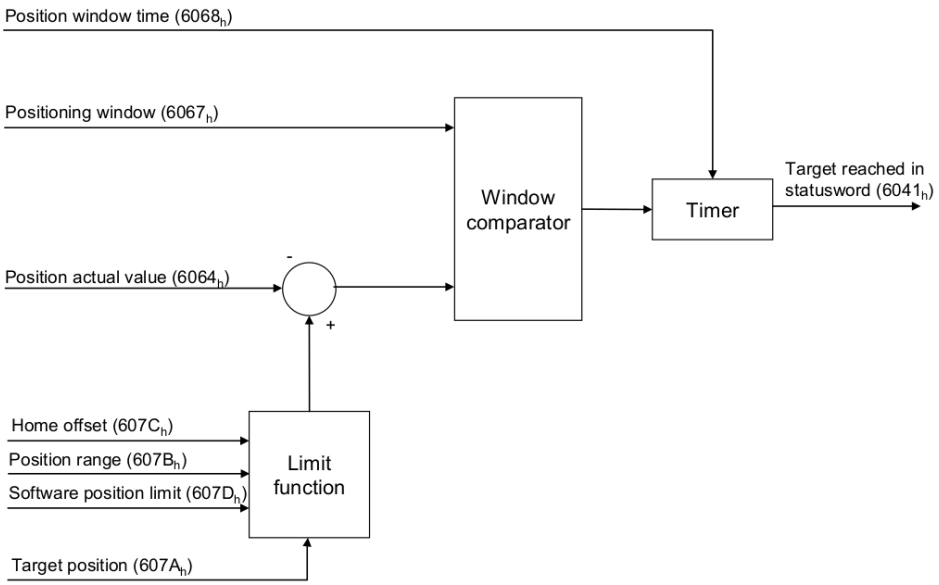
If the actual value is inside of the accepted position range for a configured period of time, the user is informed that the target position has been reached.

This information is reflected by bit 10 of the statusword.

> Use case:
>
> Pick and place operation
>
> In case the real position where the good is to be picked up or placed has more margin than the precision of the motion control system, using a wider position window allows a faster operation.

Definitions





Configuring the position window (CSP)

> Note:
>
> The accepted position range is calculated as two times the position window.

- Enter the position window value in object 0x6067 Position window (unit is given in increments of the position encoder). The value 0xFFFFFFFF switches the feature off (default).
- In object 0x6068 Position window time you can configure the time required for the position sensor to remain within the position window before it's considered to be inside the window. The unit is given in ms.

Configuring the velocity window (CSV)

> Note:
>
> The accepted velocity range is calculated as two times the velocity window.

- Enter the velocity window value in object 0x606D Velocity window (unit is given in rpm) The value 0xFFFF switches the feature off (default).
- In object 0x606E Velocity window time you can configure the time required for the value to remain within the velocity window before it's considered to be inside the window. The unit is given in ms.

Configuring the torque window (CST)

> Note:
>
> The accepted torque range is calculated as two times the torque window.

- Enter the torque window value in object 0x2014 Torque window subitem 1.
- In subitem 2 you can configure the time required for the torque value to remain within the torque window before it's considered to be inside the window. The unit is given in ms.

Result

Bit 10 of the Statusword indicates that the position is within the allowed window:

- 0: value outside accepted position range (position not reached)
- 1: position reached

Monitoring the internal variables

To examine the inputs and outputs of a given controller, the following objects can be used:

- Position demand value 0x6062: Provides the value of the position sent to the position controller after position limits are checked. (See 0x607D Software position limit.) The units are in increments of the position encoder.
- Position demand internal value 0x60FC: Provides the output of the trajectory generator in profile position mode. The units are in increments of the position encoder.
- Position value 0x6064: Provides the actual value of the position encoder. The units are in increments of the position encoder.
- Velocity demand value 0x606B: Provides the output value of the trajectory generator in velocity control, and the output of the position controller when using the cascaded controller. The units are user-defined (RPM by default; see 0x60A9 SI unit velocity to change it).
- Velocity value 0x606C: Provides the actual velocity value derived either from the velocity sensor or the position sensor. The units are user-defined (default is RPM; see 0x60A9 SI unit velocity to change it).
- Torque demand 0x6074: Provides the output value of the torque trajectory generator. The units are per thousand of rated torque.

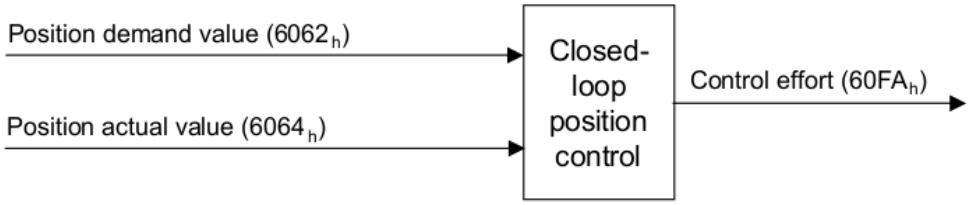
- Torque value 0x6077: Provides the actual value of the torque, and is computed based on the currents measured in the motor. The units are per thousand of rated torque.

Control Effort

The control effort (object 0x60FA) is the output of the controller for the selected mode.

It provides a measure of the control effort (the output of the position control loop). The units are in user-defined velocity units (RPM by default; see 0x60A9 SI unit velocity to change it).

- If cascaded position control is used, the control effort is velocity demand value
- If velocity control is used, the control effort is torque demand value



Parameters

Object ID	Name	Description
0x2014	Torque window	Indicates the accepted torque values and time
0x2017	Following error option codes	Configures drive's response in case of Position following error event
0x60F4	Following error actual value	Provides the actual value of the following error
0x6065	Following error window	Half of the configured range of tolerated position values
0x6066	Following error timeout	Minimum time that the actual position must be out of the window so that a following error is triggered

Object ID	Name	Description
0x6067	Position window	Half of the accepted position range
0x6068	Position window time	Time required for the position sensor to remain within the position window
0x606D	Velocity window	Half of the accepted velocity range
0x606E	Velocity window time	Time required for the value to remain within the velocity window
0x6062	Position demand value	Value of the position sent to the position controller after position limits are checked
0x60FC	Position demand internal value	Provides the output of the trajectory generator in profile position mode
0x6064	Position value	Provides the actual value of the position encoder
0x606B	Velocity demand value	Value of the trajectory generator in velocity control / output of the position controller in cascaded control
0x606C	Velocity value	Provides the actual velocity value derived either from the velocity sensor or the position sensor
0x6074	Torque demand	Provides the output value of the torque trajectory generator
0x6077	Torque value	Provides the actual value of the torque
0x60FA	Control effort	Output of the controller for the selected mode

Feed constant

When a system is performing linear motion, the values for position, velocity and acceleration can be shown in linear units.

To perform this calculation, the traveled distance per one revolution must be measured.

> Note:
> Linear direct drives are not supported! The filter is used to smooth the derivative of the target position. This is needed to soften communication lags (if present), attenuate noise in the target position signal and diminish the influence of target position quantization.

Usage

The value must be inserted in object 0x6092 (Feed constant) as follows:

Parameter	Name	Description
0x6092:1	Feed	Measured distance
0x6092:2	Shaft revolutions	Conducted revolutions

> Use case:
> During 39 revolutions, the system moved 10 mm linearly

Parameter	Name	Description
0x6092:1	Feed	10
0x6092:2	Shaft revolutions	39

> Note:
> The values can be given in real value.

In order to disable the feed, set the Parameter 0x6092:1 Feed to 0xFFFFFFFF (default value)

The user-defined position unit is defined by the Feed constant. The unit can be selected in OBLAC Drives.

If the Feed is enabled, the position unit depend on the given feed. If the feed is given in mm, then all displayed position values are also in mm.

If the Feed is disabled, the position unit is position encoder increments.

This affects all occurrences of position values in the following objects:

- Position actual value 0x6064
- Position demand internal value 0x60FC
- Target position 0x607A
- Position range limit 0x607B
- Software position limit 0x607D
- Position window 0x6067
- Following error window 0x6065
- Home offset 0x607C
- Following error actual value 0x60F4
- Homing Methods Home position 0x2005:1
- Homing Methods Offset from Endstop 0x2005:4
- Position demand value 0x6062
- Velocity and acceleration values are also affected:

Velocity and acceleration values are also affected

- Target velocity 0x60FF
- Velocity offset 0x60B1
- Velocity feedforward 0x2015
- Velocity Demand Value 0x606B
- Velocity actual value 0x606C
- Quick-stop deceleration 0x6085

- Homing acceleration 0x609A
- Homing speeds 0x6099

> Note:
>
> Max motor speed is not affected

Limitations

When the resolution in the user defined unit is larger than the resolution in encoder ticks, the drive may not move to the desired position.

Example:

- An encoder is used with a resolution of 4096 ticks
- The feed constant is given with 10 000 mm per revolution

Encoder resolution	Feed constant	Target Position value	Internal value
4096 ticks	10 000 mm / rev	1 mm	0 ticks
		500 mm	205 ticks

As a result, the motor would not move for a given command of 1 mm, as the position encoder does not have enough resolution. Ideal position demand value in that case would be

$$\frac{1*4096}{10000} = 0.4096 \text{ ticks},$$

while internal command would be 0 ticks, since ideal command is not controllable and reachable.

In the other example when target position is 500 mm, ideal demand position would be

$$\frac{500*4096}{10000} = 204.8 \text{ ticks},$$

while internal command would be 205 ticks.

Threshold

Not all user defined positions can be translated to unique ticks.
If a given target position is not reachable, the profiler will still move the shaft until the reached linear position is within a given threshold.
The threshold is defined/calculated as:

Velocity units

For defining the velocity units as inc/s:

1. set the Feed Constant to increments (position encoder resolution)
2. Set SI unit velocity data object 0x60A9 to RPS

Velocity values can be scaled between user unit and internal RPM:

- Velocity [position unit / time] = Internal velocity [RPM] * Feed constant [position unit /revolution]
- Internal velocity [RPM] = Velocity [position unit / time] / Feed constant [position unit / revolution]

Acceleration units

The user-defined acceleration units are given in [user-defined velocity unit /s]. The default unit is RPM / s (when feed disabled).

Acceleration values can be scaled between user unit and internal RPM/s:

- Acceleration [velocity_unit / s] = internal acceleration [RPM / s] * Feed constant [position_unit / rev]
- Internal acceleration [RPM / s] = Acceleration [velocity_unit / s] / Feed constant [position_unit / rev]

Errors

Some error code is provided in the error report object 0x203F:

IvldFeed: Feed is enabled and the feed constant is set to 0 or the number of revolutions is 0.

IvldPara: Multiplication of the Feed constant and SI unit velocity scaling factor overflows. This can also be caused when the acceleration value is set bigger than the maximum value (Q10 unsigned 32-bit format)

The acceleration units must fit into the internal Q10 format after scaling.

The velocity value internally is in Q15 signed 32-bit format, max. value 65 535 RPM.

If the velocity is set to a bigger value than the maximum value, the firmware doesn't prevent an overflow and internal value will be wrong.

Parameters

Parameter	Name	Description
0x6092:1	Feed	Measured distance
0x6092:2	Shaft revolutions	Conducted revolutions

Store Home Position

Overview

The home position value can be saved in the Object Dictionary. When this feature is configured, it is also stored in the configuration file each time homing is performed. After a power cycle or if the configuration needs to be reloaded, the homing value can be restored so the drive can be referenced without repeating the homing procedure.

Restoring the home value after a power cycle or config reload should be done with absolute position encoders on the output shaft (dual loop setups, multi-turn encoders with battery, single-turn encoders on torque motors).

> Note:
> Incremental encoders or setups where a gearbox is after the absolute encoders homing position should not be used for restoring the home position.

Usage

On the first power on, the position of each of the axes is random, given the mechanical mounting of the motor.

The master controller enters OPMODE HOMING and reads bit 12 (Homing Attained) of the statusword. On each axis, the value of bit 12 is zero, which means the axis is not referenced.

> Note:
> The axes need to be homed before the master allows turning on CSP

- Choose the homing method and run the homing process.
- When the axis is homed, the homing value is updated in the subindex 2005:1 Home Position and bit 12 of the statusword is set to 1 by the firmware.
- Change the parameter Restore Home Position on Startup to Yes.
- The values of the objects Home Position and Restore Home Position on Startup must be saved to the config file (for example, by clicking "Save To Device" in OBLAC Box).

After power cycling the drives, the controller goes into OPMODE HOMING and reads the value 1 in bit 12 for each drive, indicating that they don't need to be homed anymore.

Upon start-up, the firmware references the axis with the value stored in subindex 2005:1 Home Position.

Parameters

Object ID	Name	Description
0x2005:1	Home Position	Encoder value at home position
0x2005:2	Restore Home Position when loading configuration	Flag that shows if the home position saved in `0x2005:1` is valid.

Distributed clocks (DC clocks)

Overview

Synchronization via distributed clocks allows to share the same system time between the master and all slaves.

This feature allows to achieve high performance, especially with sharp tuning.

Without synchronization, unwanted vibration can occur when a slave's internal clock isn't synchronized with the master time. Especially when several slaves try to perform complex motion synchronously, periodical vibrations, audible noise and spikes in the torque signal can appear and the slaves may not execute the motion fully synchronously.

This is most prevalent in cyclic modes, since in profile modes the slaves compute their trajectories themselves.

> Important:
>
> For good performance, always enable the feature if possible. Ensure that your master supports it!

Distributed Clock configuration

SOMANET supports 2 DC modes:

- Slave mode - Master device will be used as a reference clock. All the slaves in the chain will be adjusted according to the master time.
- Master mode - One EtherCAT device will be used as a reference clock. Typically, the reference clock is the first slave with DC capability between master and all the slaves to be synchronized (DC slaves).

This feature has to be configured on the master side. We recommend user to use the Master mode as it has better performance than the Slave mode.

Additionally, the synchronization options have to be configured:

- SYNC 0 has to be enabled with recommended cycle time equal to the EtherCAT communication.
- SYNC 0 Shift time is usually 0.
- SYNC 1 is not supported yet and needs to be disabled.

Parameters

Object ID	Name	Description
0x20F3	DC synchronization	The status of the selected network synchronization mechanism

OS command

Overview

The OS command is a standard way of triggering actions or services on the drive by which cannot be adapted to a single SDO upload/download.

The following commands are currently available:

Command 0 Encoder register communication

Command 1 iC-MU calibration mode

Command 2 Open Loop Field Mode (OLFM)

Command 3 HRD streaming

Command 4 Motor phase order detection

Command 5 Commutation offset detection

Command 6 Open phase detection

Command 7 Pole pair detection

Command 8 Phase resistance measurement

Command 9 Phase inductance measurement

Command 10 Torque constant measurement

Command 13 Skipped cycles counter

Command 14 Ignore BISS status bits

Usage

The OS command feature can be operated with the objects 0x1023 (OS command) and 0x1024 (OS command mode).

0x1023: OS command

Select the command that should be used and start it by writing the desired value in subindex 1.

The status can be read in subindex 2.

The response can be fetched in subindex 3, it includes the status as written in subindex 2.

> Note:
>
> To calculate the progress, please subtract 100 from the value of 0x1023:2.

Subindex	Description	Data Type	Value
1	Command	OCTET_STRING	**Byte 0**: OS command ID **Byte 1-7**: Service request data A write access to the subindex will execute the command
2	Status	UNSIGNED8	**0**: last command completed, no errors, no response data **1**: last command completed, no errors, response data available **2**: last command completed, error, no response data **3**: last command completed, error, response data available **100-200**: command executing with percentage (100 = 0%, 200 = 100%) **255**: command executing (if percentage display is not supported)
3	Response	OCTET_STRING	**Byte 0**: identical with subindex 2 **Byte 1**: unused **Byte 2-7**: Service response data

Details

- Subindex 1 can't be written until the command is finished and the response is read in subindex 3. If a command is written in that case anyway, it will be ignored and an OsCmdCol error will be raised.
- The status of the command can be read in Subindex 2 and is also available in the first byte of subindex 3.

> Note:
> It is recommended to read the status from the first byte of subindex 3 instead of subindex 2 because reading subindex 2 when a command is finished will not make subindex 1 writable again. The subindex 3 has to be read at least once after writing, before the next write command can be performed.

- Both subindex 1 and subindex 3 are octet strings of size 8.
- Byte 0 of the subindex 1 is used to choose the command that shall be executed.
- The rest of the bytes are used to specify the parameters of the command, so they are command specific.
- Byte 0 of the subindex 3 shows the status of the command. Depending on its value the rest of bytes on that subindex differ:
    - If the command is in progress (byte 0 is 100-200 or 255) or completed with no response data (byte 0 is 0 or 2), no other bytes will be set.
    - If the command is completed with no errors and with response data (byte 0 is 1), command specific bytes will be set starting at byte 2.
    - If the command is completed with errors and with response data (byte 0 is 3), byte 2 will be the OS error code (see below) and command specific bytes will be set starting at byte 3.

The following table shows all the valid responses (subindex 3) that any OS command might respond with. Grey table cells mean: these bytes are optional.

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (with percentage)	100 - 200							
Command in progress (without percentage)	255							
Command completed without errors and without response	0							
Command completed without errors and with	1		Data	Data	Data	Data	Data	Data

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
response								
Command completed with error and without response	2							
Command completed with error and response	3		OS error code (see below)	Data	Data	Data	Data	Data

OS error code

Indicates the reason for an error when the current command is completed with errors and with response data (status is 3).

> Note:
> Error codes 0 - 8 are command specific, while error codes 251 - 254 are general error codes.

The command-specific OS error codes will be explained in detail in the corresponding section.

General OS error codes:

- Unsupported command: the command ID received doesn't exist.
- Command timeout: A request to execute an OS command or to abort it was not acknowledged by any downstream service after 5s.
- Command aborted: The abort action triggered by the user was successful.
- Command not allowed: Certain commands can execute only under certain conditions (in a specific operation mode for example). This error code will be used to signal that these conditions are not met.

	OS Command #										
	0	1	2	3	4	5	6	7	8	9	10
OS error code 0	Invalid response				Initialization error			Open terminal A	Current ampl. error		
OS error code 1	Register not allowed				Streaming error			Open terminal B			
OS error code 2					Duration value			Open terminal C			

OS Command #											
	0	1	2	3	4	5	6	7	8	9	1-0
OS error code 3					Data index value			Open FET A high			
OS error code 4					Action value			Open FET A low			
OS error code 5								Open FET B high			
OS error code 6								Open FET B low			
OS err-								Open			

OS Command #											
	0	1	2	3	4	5	6	7	8	9	1-0
or co-de 7								FET C high			
OS err-or co-de 8								Ope-n FET C low			
...		(reserved for future use)									
OS err-or co-de 25-1		Command not allowed									
OS err-or co-de 25-2		Command aborted									

	OS Command #									
	0	1	2	3	4	5	6	7	8	9	1-0
OS error code 25-3		Command timeout									
OS error code 25-4		Unsupported command									

0x1024: OS command mode

This object is only used to abort the current command when value 0x03 is written. Otherwise (value 0x00 is written as a default) the next command is executed immediately.

Value	Description
0x00	Execute the next command immediately (default)
0x03	Abort the current command and all commands in the buffer

Details

- This object is write-only.
- Writing 0 to this object will enable “Execute the next command immediately”, which is the default mode. In this mode, new OS commands will be executed as described above.

- Writing 3 to this object will abort the currently executing command. Once the command is aborted successfully, the status “last command completed, error, response data available” (3) will be sent by the slave, and the OS error code (which will be explained in more detail later) in byte 2 of the response will signal the command abortion. If there is no command in progress, this write operation will be ignored.

OS command behavior

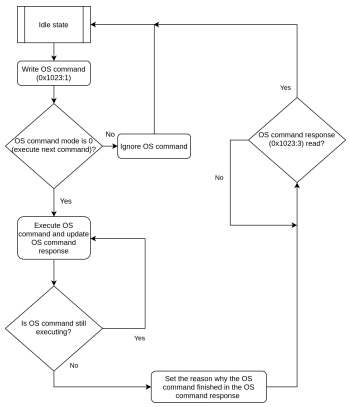
The drive only accepts OS commands when the OS command state is idle, which means that the last command was completed and the response was read from 0x1023:3. Otherwise the OS command will be ignored, and the drive will enter the Fault CiA402 state (error report description will be OsCmdCol, which stands for OS command collision).

If subindex 0x1024:0 is not 0, the OS command will be ignored and discarded and no error will be raised.

Whenever an OS command is sent, the status will change from 0, 1, 2 or 3 to 255 immediately to ensure that the correct response is received.

At any point while a command is in progress, that command can be aborted writing 3 to the object 0x1024.

Flowchart of the Command



Available commands

Command 0: Encoder register communication

This command is used for reading or writing registers from a BiSS encoder.

> Note:
> For more information about how BiSS register communication works, please refer to the official BiSS-C Interface Protocol Description.

Command request

The values in 0x1023:1 for this command are as follows:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0							
1						Encoder ordinal		
2	Slave address / ID (0)							¬R/W
3	Register address							
4	Register write value							

Explanation of the parameters:

- Encoder ordinal: select the target encoder for the command.
  A value of 1 will select the Encoder 1 (which is the encoder configured in 0x2110), a value of 2 will select Encoder 2 (which is the encoder configured in 0x2112).

Value	Encoder
1	Encoder 1
2	Encoder 2

- Slave address: selects the address of the slave to read (BiSS protocol level). This value must always be 0.
- ¬R/W: Selects the direction of the command: 0 for reading a register, 1 for writing it.
- Register address: Register to be read/written.

- Register write value: Value to write to the register. This is only used if the parameter ¬R/W is set to 1.

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without errors and with response	1		Register value					
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The register value is the value read/ written from/to the selected register.

- **Status 3**: The command has finished unsuccessfully. The error code will be sent, either as a general OS error code or a command-specific OS error code:

OS error code	Error name	Error description
0	Invalid response	There was an error while executing the BiSS register transaction. This can be caused by • a too short BiSS timeout, • too high BiSS clock frequency or • trying to access a register that does not exist
1	Register not allowed	The register can't be read given the current configuration. Currently the only registers that are not allowed to read are the iC-MU I2C registers (range 0x5B-0x6F) in Internal encoder 1 of Circulo if multiturn is disabled. If this happens, it can be avoided by setting the subindex "Status bits active value" to "Status bits ignored".

Example

To write the value 0x08 in register 0x11 of the encoder configured as Encoder 2, the bytes in 0x1023:1 should be set like this:

Byte	Value
0	0
1	2
2	1
3	0x11
4	0x08

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0811010200 --type int64
```

> Note:
>
> This command can only be used with IgH ethercat Master for Linux

To fetch the response, the following command should be entered:

```
ethercat upload 0x1023 3 --type int64
```

This value should be returned: 0x0000000000080001 (Status 1, with a confirmation of the written register value).

Command 1: iC-MU calibration mode

This command is used to be able to set a particular BiSS mode when using an iC-MU encoder, which is mostly used for iC-MU calibration. The following modes are supported:

- Configuration mode: the BiSS service reads the encoder normally but it only uses the bits for register communication and discards the rest, so the encoder data is not updated. This also means *BisCrc* error is not raised while in this mode.
- Raw mode: The BiSS service reads the encoder assuming it was correctly configured for raw mode. It continues calculating all the encoder data but in addition to that it also average of the raw data and makes it available in object 0x2704 User MISO like this:

Bit		
31-28	27-14	16-0
0	Nonius count	Master count

- Standard mode: The BiSS service clocks the encoder normally.

When transitioning from any mode to Configuration mode, the current position is saved. When switching to Raw mode, this saved position will be used as a starting point for the position calculation, because the raw data is relative. This means that the motor should not be moved while in Configuration mode if we plan to switch to Raw mode after.

Command request

Values in 0x1023:1 for this command:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	1							
1				Mode enable		Encoder ordinal		

Explanation of the parameters:

- Mode enable: select the iC-MU calibration mode.

Value	Mode
0	Configuration mode
1	Raw mode
2	Standard mode

- Encoder ordinal: select the target encoder for the command.

  A value of 1 will select the Encoder 1 (which is the encoder configured in 0x2110), a value of 2 will select Encoder 2 (which is the encoder configured in 0x2112).

Value	Encoder
1	Encoder 1
2	Encoder 2

Command response

Possible responses of this command:

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without per-	255							

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
centage)								
Command completed without errors and without response	0							
Command completed with error and without response	2							

- **Status 255**: The command is in progress.
- **Status 0**: The command was finished successfully.
- **Status 2**: The command couldn't be executed on the sensor. The received Mode enable value is unknown.
- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

Configuring raw mode on encoder configured as Encoder 1:

1. Enable Configuration mode to freeze the current absolute position.

   To change Internal encoder 1 of Circulo to Configuration mode, the bytes in 0x1023:1 need to be set like this:

Byte	Value
0	1
1	1

   In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0101 --type int64
```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

   Which should return 0x0000000000000000 (Status 0).

2. Use OS command 0 to configure Encoder 1 to output raw mode. Check the iC-MU data sheet for more information.

3. Enable Raw mode.

   To change Encoder 1 to Raw mode, the bytes in 0x1023:1 need to be set like this:

Byte	Value
0	1
1	9

   In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0901 --type int64
```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

   Which should return 0x0000000000000000 (Status 0).

Command 2: Open Loop Field Mode (OLFM)

This command is used to be able to configure the settings of the Open Loop Field Mode (OLFM) profiler. The following configuration can be set by using this OS command:

Parameters related to the profiler of the electrical angle:

- Starting electrical angle [mrad]
- Ending electrical angle [mrad]
- Maximum electrical rotational speed [rad / s]
- Electrical rotational acceleration [rad / s²]

Parameters related to the profiler of the magnetic field length:

- Starting length [per thousand rated current]
- Ending length [per thousand rated current]
- Increasing length rate [per thousand rated current / s]

Once the configurations of OLFM profiler are set, the user can put the drive in the corresponding OLFM op_mode (The op_mode for OLFM operation is -3), and by enabling the operation the Drive will generate a magnetic field according to the settings of its profiler. More information about this mode can be found in [Missing link].

Command request

Values in 0x1023:1 for this command:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	2							
Byte 1	Parameter index							
Byte 2	Parameter value [MSB]							
Byte 3	Parameter value [MSB-1]							
Byte 4	Parameter value [MSB-2]							
Byte 5	Parameter value [LSB]							

Explanation of the parameters

- Parameter index: Selects the parameter that shall be configured.

Value	Parameter and units
0	Starting electrical angle [mrad]
1	Ending electrical angle [mrad]
2	Maximum electrical rotational speed [rad / s]
3	Electrical rotational acceleration [rad / s²]
4	Starting length [per thousand rated current]
5	Ending length [per thousand rated current]
6	Increasing length rate [per thousand rated current / s]

- Parameter value: The value to set the selected parameter to. The maximum size for the value is 32 bits and Byte 5 is the least significant byte (LSB).

Command response

The possible responses of this command:

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command	0							

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
completed without errors and without response								
Command completed with error and without response	2							
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 0**: The command was finished successfully.
- **Status 2**: The command couldn't be executed. The received Parameter index value is unknown.

- **Status 3**: This command doesn't have any command-specific OS error code (this status will only happen when a general OS error code occurs).

Example

For setting the "Starting electrical angle [mrad]" to 1000, the parameter value should be converted to hexadecimal so that it is easier to split between multiple bytes. In this case, 1000 is 0x3E8.

Then we should set the bytes in 0x1023:1 like this:

Byte	Value
0	2
1	0
2	0
3	0
4	0x03
5	0xE8

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0xe80300000002 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

Which should return 0x0000000000000000 (Status 0).

Command 3: HRD streaming

This command is used for streaming data into files. The data written to these files is in binary format.

- Multiple files will be created if necessary, up to a maximum of 5 files.
- Each of the files will have a maximum size of 8032 bytes
- The files are named "hr_dataX.bin", where X is a number starting at 0.

Two actions are supported for this command:

- Configure stream: all previous HRD files are deleted from the filesystem, and the user configures the duration of the stream and the variable to stream. Deleting files can take up to 5 seconds in the worst case scenario.
- Start stream: The stream is started with the previously sent configuration.

Command request

The values in 0x1023:1 for this command are as follows:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	3							
1	-							Action
2	Data index							
3	Duration in ms [MSB]							
4	Duration in ms [LSB]							

Explanation of the parameters:

- Action: select the action to perform.

Value	Action
0	Configure stream
1	Start stream

- Data index: select the data to be streamed. This value is only used when the selected Action is Configure stream (0).

Value	Data	Description
0	Encoder raw data	Record encoder raw data. The encoder should be set up first with OS command 1.
1	System iden-	Record system identification data. The identification

Value	Data	Description
	tification data	procedure should be configured with OS command 15 and started before recording of the data.

- Duration in ms: select the duration in milli-seconds of the stream. This value is only used when the Action is Configure stream (0). The maximum value for this parameter is 10000 ms.

Command response

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (with percentage)	100 - 200							
Command in progress (without percentage)	255							
Command completed without errors and	0							

Description	Byte 0 (Status)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
without response								
Command completed with error and response	3		OS error code					

- **Status 100-200**: The command to stream the data to a file (Action is 1) is in progress.
- **Status 255**: The command to configure the stream (Action is 0) is in progress.
- **Status 0**: The command has finished successfully.
- **Status 3**: The command has finished unsuccessfully. The error code will be sent, either as a
  - general OS error code or a
  - command-specific OS error code:

OS error code	Error name	Error description
0	Initialization error	The command failed to initialize.
1	Streaming error	Something failed while streaming was in progress.
2	Duration value	The selected Duration value is larger than 10000 ms.
3	Data index value	The selected Data index is unknown.
4	Action value	The selected Action is unknown.

Example

Streaming encoder raw data into a file for 5 s:

1. Use OS command 0 and OS command 1 to configure the iC-MU to output raw data.
2. Configure the streaming:

   To configure the stream first we need to convert the duration to hexadecimal so that it is easier to split between multiple bytes. In this case, 5000 is 0x1388.

   The bytes in 0x1023:1 should be:

Byte	Value
0	3
1	0
2	0
3	0x13
4	0x88

   In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x8813000003 --type int64
```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

   Which should return 0x0000000000000000 (Status 0).

3. Start the streaming.

   The bytes in 0x1023:1 should be:

Byte	Value
0	3
1	1

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x0103 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

Which should return 0x0000000000000000 (Status 0).

Command 4: Motor phase order detection

This command is used to be able to detect the order of the motor phases, it is also the first step that is performed during Offset detection. When this command finishes successfully the found value is automatically written to the object 0x2003:5 Motor phases inverted.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically finished. More information about this operation mode can be found in the Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	4							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Com-	1		Motor					

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
mand com-pleted without errors and with response			phas-e order					
Com-mand com-pleted with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The detected motor phase order is written in byte 2.

Value	Motor phase order
0	Normal order
1	Inverted order

- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

To measure the motor phase order of our setup, we would need to:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one. For details please refer to Diagnostics opmode.
2. Execute the OS command.

Set Byte 0 of 0x1023:1 to 4.

In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x04 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

- If the Motor phase order is normal, the response would be 0x0000000000000001 (Status 0, Normal motor phase order).
- If the Motor phase order is inverted, the response would be 0x0000000000010001 (Status 1, Inverted motor phase order).

Command 5: Commutation offset detection

This command is used to measure the commutation angle offset which is the second step in the Offset Detection procedure. The configuration of the procedure is taken from the already existing and previously used commutation offset objects in the object dictionary. When this command finishes successfully the found value is automatically written to the object 0x2001:0 Commutation angle offset, and the object 0x2009:1 State is set to OFFSET_VALID.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. More information about this operation mode can be found in Diagnostics opmode.

For details about the different methods for Offset detection, refer to Commutation Offset Detection.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	5							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without errors and with response	1		Commutation angle offset [MSB]	Commutation angle offset [LSB]				
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured angle offset is written in bytes 2 and 3.

- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

To measure the commutation angle offset of our setup, we would need to:

1. Configure the Commutation offset detection parameters in the object 0x2009.
2. Switch to diagnostics mode, enter operation enable and disengage the brake if necessary. For details please refer to Diagnostics opmode.
3. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 5. In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x05 --type int64
   ```

   To fetch the response, enter the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

   In case the Commutation angle offset is 2035, this would return 0x0000000007F30001 (Status 1, 2035 is 0x7F3 in hexadecimal).

Command 6: Open phase detection

This command is used for detecting if one of the phases is open.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. For details please refer to Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	6							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without errors and without response	0							
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 0**: The command was finished successfully, no open motor phase was detected.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Open terminal A	Terminal A of the Drive is not connected
1	Open terminal B	Terminal B of the Drive is not connected
2	Open terminal C	Terminal C of the Drive is not connected
3	Open FET A high	Upper FET in leg A of the Drive is not conducting (open circuit fault)
4	Open FET A low	Lower FET in leg A of the Drive is not conducting (open circuit fault)
5	Open FET B high	Upper FET in leg B of the Drive is not conducting (open circuit fault)
6	Open FET B low	Lower FET in leg B of the Drive is not conducting (open circuit fault)
7	Open FET C high	Upper FET in leg C of the Drive is not conducting (open circuit fault)
8	Open FET C low	Lower FET in leg C of the Drive is not conducting (open circuit fault)

Example

To detect if any of the motor phases is open in our setup, we would need to:

1. Switch to diagnostics mode and enter operation enable.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 6. In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x06 --type int64
   ```

   To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

- An OS error code will indicate if there is an issue
- For example if terminal B is open this would return 0x0000000000010003 (Status 3, OS error code 1).
- If no issue was found, this would return 0x0000000000000000 (Status 0).

Command 7: Pole pair detection

This command is used to detect the amount of pole pairs that the connected motor has.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. For details please refer to Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	7							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without	1		Number of pole pairs					

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
errors and with response								
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured number of motor pole pairs is written in byte 2.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Current ampl. error	Drive is not able to increase the amplitude of motor phase currents to the required value. This problem can happen due to limited V_{DC} or high motor phase impedances

Example

For measuring the motor phase order:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 7. In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x07 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

In case the motor has 3 pole pairs, this would return 0x0000000000030001 (Status 1).

Command 8: Phase resistance measurement

This command is used to measure the phase resistance of the motor in mOhm.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. For details please refer to Diagnostics opmode.

Command request

Values in 0x1023:1 for this command:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	8							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without	1		Phase resistance [MSB]	Phase resistance [MSB-	Phase resistance [MSB-	Phase resistance [LSB]		

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
errors and with response				1]	2]			
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured motor phase resistance is written in bytes 2, 3, 4 and 5.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Current ampl. error	Unable to increase the current amplitude.

Example

To measure the motor phase resistance of our setup, we would need to:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 8. In the terminal the following command can be used:

```
ethercat download 0x1023 1 0x08 --type int64
```

To fetch the response, enter the following command:

```
ethercat upload 0x1023 3 --type int64
```

In case the motor phase resistance is 100000, this would return 0x0000000186A00001 (Status 1, 100000 is 0x186A0 in hexadecimal).

Command 9: Phase inductance measurement

This command is used to measure the phase inductance of the motor in uH.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. More information about this operation mode can be found in the Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	9							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Com-	1		Phase	Phase	Phase	Phase		

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
mand completed without errors and with response			inductance [MSB]	inductance [MSB-1]	inductance [MSB-2]	inductance [LSB]		
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured motor phase inductance is written in bytes 2, 3, 4 and 5.
- **Status 3**: The command was finished unsuccessfully. Either a general OS error code or a command-specific OS error code will be sent.

OS error code	Error name	Error description
0	Current ampl. error	Unable to increase the current amplitude.

Example

To measure the motor phase inductance of our setup, we would need to:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one.
2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 9. In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x09 --type int64
   ```

   To fetch the response, enter the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

   In case the motor phase inductance is 4000, this would return 0x0000000010940001 (Status 1, 4244 is 0x1094 in hexadecimal).

Command 10: Torque constant measurement

This command is used to measure the torque constant of the motor in mNm/A $_{RMS}$.

This command can only be run when the operation mode is -2 (OPMODE_DIAGNOSTICS) and the CiA402 state is Operation enabled. If any of these conditions is not true during the execution of the command, it will be automatically aborted. More information about this operation mode can be found in Diagnostics opmode.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	0x0A							

Command response

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without	255							

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
percentage)								
Command completed without errors and with response	1		Torque constant [MSB]	Torque constant [MSB-1]	Torque constant [MSB-2]	Torque constant [LSB]		
Command completed with error and response	3		OS error code					

- **Status 255**: The command is in progress.
- **Status 1**: The command was finished successfully. The measured motor phase inductance is written in bytes 2, 3, 4 and 5.
- **Status 3**: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

For measuring the motor torque constant:

1. Switch to diagnostics mode, enter operation enable and disengage the brake if there is one. For details please refer to Diagnostics opmode.

2. Execute the OS command.

   Set Byte 0 of 0x1023:1 to 10 (0xA). In the terminal the following command can be used:

   ```
   ethercat download 0x1023 1 0x0A --type int64
   ```

   To fetch the response, enter the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

   In case the motor torque constant is 1203210, this would return 0x000000125C0A0001 (Status 1, 1203210 is 0x125C0A in hexadecimal).

Command 13: Skipped cycles counter

This command is used to check the number of firmware cycles skipped. It provides the value of an internal counter that keeps track of the number of skipped cycles.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	0x0D							
Byte 1	Firmware service							

Explanation of the parameters:

- Firmware service: Select in which service the action should be performed.

Value	Firmware Service
0	Drive control service
1	Motion control service

Command response

The possible responses that this command will use are the following:

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255	-	-	-	-	-	-	-
Command completed without errors and with response	1	-	Number of skipped cycles [MSB]	Number of skipped cycles [MSB-1]	Number of skipped cycles [MSB-2]	Number of skipped cycles [LSB]	-	-
Command completed with error and response	3	-	OS error code	-	-	-	-	-

- Status 255: The command is in progress.
- Status 1: The command finished successfully. The number of skipped cycles is written in bytes 2, 3, 4 and 5.
- Status 3: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

To get the value of the skipped cycles counter of Motion control service:

1. Set Byte 0 of 0x1023:1 to 13 (0xD), and Byte 1 to 1.

   In the terminal we can use the following command:

   ```
   ethercat download 0x1023 1 0x010D --type int64
   ```

   To fetch the response, we should type the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

This will return a value, for example 23. If we run the command again after some time and we get for example 33, this will mean that 10 cycles were skipped in the timespan between both OS commands.

Command 14: Ignore BiSS status bits

This command is used to ignore the BiSS status bits, so that they don't raise a firmware warning or error.

In the case of Circulo, it also stops reading the iC-PVL STATUS register when multiturn is enabled.

Command request

The values in 0x1023:1 for this command are as follows:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	0x0E							
Byte 1						Encoder ordinal		Trigger

Explanation of the parameters:

- Trigger: a value of 1 will ignore the status bits. A value of 0 will stop ignoring them, which is the default behavior of the firmware.
- Encoder ordinal: select the target encoder for the command. A value of 1 will select Encoder 1 (which is the encoder configured in 0x2110: Encoder 1 configuration), a value of 2 will select Encoder 2 (which is the encoder configured in 0x2112: Encoder 2 configuration).

Value	Encoder
1	Encoder 1
2	Encoder 2

Command response

The possible responses that this command will use are the following:

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Command in progress (without percentage)	255							
Command completed without errors and without response	0							
Command com-	3		OS error code					

Description	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
pleted with error and response								

- Status 255: The command is in progress.
- Status 0: The command finished successfully.
- Status 3: This command doesn't have any command-specific OS error code, so this status will only happen when a general OS error code occurs.

Example

If we want to ignore the BiSS status bits of Encoder 1 we should:

1. Set Byte 0 of 0x1023:1 to 14 (0xE), and Byte 1 to 3 (0b11). In the terminal we can use the following command:

   ```
   ethercat download 0x1023 1 0x030E --type int64
   ```

   To fetch the response, we should type the following command:

   ```
   ethercat upload 0x1023 3 --type int64
   ```

Parameters

Object ID	Name	Description
0x1023	OS command	Selects, starts and reports the status of the command that should be used
0x1024	OS command mode	Orders the next command to be processed or aborts all commands

Quick stop

Overview

When the drive is in motion and the user wishes to stop quickly for any reason, the Quick Stop state can be activated. This function uses a profile to drive the motor from the current velocity down to zero with a specific Quick Stop deceleration. Quick stop can also be performed when an internal fault happens to stop the drive in a controlled way.

> Attention:
> Note that the brake might get damaged when the motor does not ramp down to zero velocity. This can happen due to one of the following reasons:

- when the motor is mechanically unstable
- when the control system is unstably or poorly tuned
- when the quick-stop feature is misconfigured

If no brake exists, this can even lead to a damaged motor!

> Note:
> At the end of the quick stop procedure, the brake gets engaged when the drive is in SWITCH_ON_DISABLED (mode 2). When the drive stays in QUICK_STOP_ACTIVE (mode 6) the brake is not engaged by default.

Usage

Quickstop will be enabled if one of the following conditions occurs:

- Bit 2 (quick stop bit) of the Controlword changed from 1 to 0 (transitions 7, 10, 11 according to the drive state machine diagram.
- A drive fault is triggered
- The drive dropped from EtherCAT Operational state while it was in CiA402 Operational Enabled state

Object 0x605A Quick stop option code has different options that switch the drive into different states:

Option code	Definition
0	Disable drive function (no quick stop deceleration)
2 (default)	Slow down on quick stop ramp and transit into switch on disabled
6	Slow down on quick stop ramp and stay in quick stop active

The execution of the procedure depends on the operational mode that is active:

Position and velocity control modes

- In position control modes (Cyclic Synchronous Position mode, Profile Position mode, Cogging recording, Homing) the quick stop procedure uses the position controller to stop the motor.
- In velocity control modes (Profile velocity mode, Cyclic Synchronous Velocity Mode), the quick stop procedure uses the velocity profiler to stop the motor.

During the procedure, torque offset can be used to compensate for possible torque from the load.

The steps vary depending on the quick stop option code and when quick stop is triggered:

Option code: 0 (Disable the drive)

> Attention:
> In this case, the drive does not perform the quick stop deceleration ramp and is instead immediately disabled.

Option code: 2 (Slow down on quick stop ramp and transit into switch on disabled)

1. The Drive commands a quick stop ramp that ramps down an effective commanded velocity according to the user-defined quick stop deceleration.
2. The end of the quick stop ramp is a moment when the commanded velocity is equal to 0.
3. A timeout period of 100 ms is started in which the drive commands a fixed position or zero velocity.

4. After the end of the quick stop ramp, the drive is disabled, which automatically engages the brake as defined by the brake settings.

Option code: 6 (Slow down on quick stop ramp and stay in quick stop active)

1. The Drive commands a quick stop ramp that ramps down an effective commanded velocity according to the user-defined quick stop deceleration.
2. The end of the quick stop ramp is a moment when the commanded velocity is equal to 0.
3. After the end of the quick stop ramp, the drive continues to stay in Quick Stop Active until a command to disable the drive is received. In position control modes (Cyclic Synchronous Position mode, Profile Position mode, Cogging recording, Homing), Quick Stop Active drive keeps the current position.

   In velocity control modes (Profile velocity mode, Cyclic Synchronous Velocity Mode), the drive controls 0 velocity.

> Attention:
>
> The brake is not engaged by default in this state.

Torque control mode

In torque modes (Cyclic Synchronous Torque mode, Profile torque mode) the quick stop procedure does not use a profile. Instead, the drive commands 0 torque from the start of the quick stop procedure. The steps vary depending on the quick stop option code and when quick stop is triggered:

Option code: 0 (Disable the drive)

> Note:
>
> In this case, the drive does not perform the quick stop deceleration ramp and is instead immediately disabled.

Option code: 2 (Slow down on quick stop ramp and transit into switch on disabled)

1. The Drive commands Target and offset torque
2. At the same moment, the timeout period starts. The timeout period is calculated as actual motor velocity divided by the quick stop deceleration. In this way, a user can configure the maximum duration of the quick stop procedure.
3. During the timeout period, the drive monitors if the drive reaches zero velocity
4. The brake gets engaged and the drive is set to state Switch On Disabled, if one of the conditions are true:
    - The actual velocity of the motor reached 0
    - The timeout period has expired

Option code: 6 (Slow down on quick stop ramp and stay in quick stop active)

1. The Drive commands Target and offset torque. According to the DS 402 standard, the drive remains in the torque control mode it was triggered from.
2. The drive continues to stay in Quick Stop Active until a command to disable the drive is received.

> Note:
> When quick stop is commanded during Offset detection, the procedure is aborted and the drive is disabled.

Parameters

Object ID	Name	Description
0x605A	Quick stop option code	Select the action that is performed when the quick stop function is executed
0x6085	Quick stop deceleration	Defines the quick stop ramp
0x6040	Controlword	Shows the drive's transition into a different state. Use this object to monitor the drive's behavior

Object ID	Name	Description
0x60A9	SI unit velocity	This object defines the velocity value
0x60FC	Position demand internal value	Provides the output of the trajectory generator in profile position mode
0x6064	Position actual value	Provides the actual value of the position encoder
0x606B	Velocity demand value	Output of position controller in position mode/trajectory generator in velocity mode
0x606C	Velocity actual value	Provides the value read at the velocity sensor (or the derivation of the position signal)
0x6074	Torque demand	Provides the output value of the torque trajectory generator
0x6077	Torque actual value	Provides the actual value of the torque

CiA 402 Object Dictionary

All Objects	A list of the entire CiA 402 Object Dictionary
Communication Area	Objects of the Communication Area
Standard RxPDO mapped Objects	The Standard mapping of Receive Process Data Objects
Standard TxPDO mapped Objects	The Standard mapping of Transmit Process Data Objects
Manufacturer Specific Area	A list of all Manufacturer Specific Objects
Profile Specific Area	A list of all Profile Specific Objects

List of all Objects

Table Caption Outside Table: Communication Area

Index		Name
0x1000		Device type
0x1001		Error register
0x1005		COB-ID SYNC
0x1006		Communication cycle period
0x1008		Manufacturer device name
0x100A		Manufacturer software version
0x100C		Guard time
0x100D		Life time factor
0x1010		Store parameters
	0x1010:1	Save all parameters
0x1011		Restore default parameters
	0x1011:1	Restore all default parameters
	0x1011:2	Restore communication default parameters (disabled)
	0x1011:3	Restore application default parameters (disabled)
	0x1011:4	Restore manufacturer default parameters
0x1016		Consumer heartbeat time
	0x1016:1	Consumer heartbeat time
0x1017		Producer heartbeat time
0x1018		Identity object

	0x1018:1	Vendor-ID
	0x1018:2	Product Code
	0x1018:3	Revision number
	0x1018:4	Serial number
0x1019		Synchronous counter overflow value
0x1023		OS command
	0x1023:1	Command
	0x1023:2	Status
	0x1023:3	Response
0x1024		OS command mode
0x1400		Receive PDO1 parameter
	0x1400:1	COB-ID used by PDO
	0x1400:2	Transmission type
	0x1400:3	Inhibit time
0x1401		Receive PDO2 parameter
	0x1401:1	COB-ID used by PDO
	0x1401:2	Transmission type
	0x1401:3	Inhibit time
0x1402		Receive PDO3 parameter
	0x1402:1	COB-ID used by PDO

	0x1402:2	Transmission type
	0x1402:3	Inhibit time
0x1403		Receive PDO4 parameter
	0x1403:1	COB-ID used by PDO
	0x1403:2	Transmission type
	0x1403:3	Inhibit time
0x1800		Transmit PDO1 parameter
	0x1800:1	COB-ID used by PDO
	0x1800:2	Transmission type
	0x1800:3	Inhibit time
	0x1800:4	reserved
	0x1800:5	Event timer
0x1801		Transmit PDO2 parameter
	0x1801:1	COB-ID used by PDO
	0x1801:2	Transmission type
	0x1801:3	Inhibit time
	0x1801:4	reserved
	0x1801:5	Event timer
0x1802		Transmit PDO3 parameter
	0x1802:1	COB-ID used by PDO

	0x1802:2	Transmission type
	0x1802:3	Inhibit time
	0x1802:4	reserved
	0x1802:5	Event timer
0x1803		Transmit PDO4 parameter
	0x1803:1	COB-ID used by PDO
	0x1803:2	Transmission type
	0x1803:3	Inhibit time
	0x1803:4	reserved
	0x1803:5	Event timer
0x1C00		Sync Manager
	0x1C00:1	SyncMan 0
	0x1C00:2	SyncMan 1
	0x1C00:3	SyncMan 2
	0x1C00:4	SyncMan 3
0x1C10		SM 0 Assignment
0x1C11		SM 1 Assignment
0x1C12		SM 2 Assignment
	0x1C12:1	SubIndex 001
	0x1C12:2	SubIndex 002

	0x1C12:3	SubIndex 003
	0x1C12:4	SubIndex 004
0x1C13		SM 3 Assignment
	0x1C13:1	SubIndex 001
	0x1C13:2	SubIndex 002
	0x1C13:3	SubIndex 003
	0x1C13:4	SubIndex 004

Table Caption Outside Table: Standard RxPDO mapped Objects

Index		PDO Index	Name
0x1600			Receive PDO1 mapping
	0x1600:1	0x6040	Controlword
	0x1600:2	0x6060	Modes of operation
	0x1600:3	0x6071	Target Torque
	0x1600:4	0x607A	Target position
	0x1600:5	0x60FF	Target velocity
	0x1600:6	0x60B2	Torque offset
	0x1600:7	0x2701	Tuning command
0x1601			Receive PDO2 mapping
	0x1601:1	0x60FE	Physical outputs
	0x1601:2	0x60FE	Bit mask

0x1602			Receive PDO3 mapping
	0x1602:1	0x2703	User MOSI
	0x1602:2	0x60B1	Velocity offset
0x1603			Receive PDO4 mapping

Table Caption Outside Table: Standard TxPDO mapped Objects

Index		PDO Index	Name
0x1A00			Transmit PDO1 mapping
	0x1A00:1	0x6041	Statusword
	0x1A00:2	0x6061	Modes of operation display
	0x1A00:3	0x6064	Position actual value
	0x1A00:4	0x606C	Velocity actual value
	0x1A00:5	0x6077	Torque actual value
0x1A01			Transmit PDO2 mapping
	0x1A01:1	0x2401	Analog input 1
	0x1A01:2	0x2402	Analog input 2
	0x1A01:3	0x2403	Analog input 3
	0x1A01:4	0x2404	Analog input 4
	0x1A01:5	0x2702	Tuning status
0x1A02			Transmit PDO3 mapping
	0x1A02:1	0x60FD	Digital inputs

0x1A03			Transmit PDO4 mapping
	0x1A03:1	0x2704	User MISO
	0x1A03:2	0x20F0	Timestamp
	0x1A03:3	0x60FC	Position demand internal value
	0x1A03:4	0x606B	Velocity demand value
	0x1A03:5	0x6074	Torque demand

Table Caption Outside Table: Manufacturer Specific Area

Index		Name
0x2000		Command object (disabled)
0x2001		Commutation angle offset
0x2002		Position control strategy
0x2003		Motor specific settings
	0x2003:1	Pole pairs
	0x2003:2	Torque constant
	0x2003:3	Phase resistance
	0x2003:4	Phase inductance
	0x2003:5	Motor phases inverted
0x2004		Brake options
	0x2004:1	Pull voltage

	0x2004:2	Hold voltage
	0x2004:3	Pull time
	0x2004:4	Release strategy
	0x2004:5	Controller disable delay
	0x2004:6	DC bus voltage (deprecated)
	0x2004:7	Brake status
	0x2004:8	Minimum displacement (pin brake)
	0x2004:9	Maximum torque (pin brake)
	0x2004:10	Output voltage
	0x2004:11	Switching frequency
0x2006		Protection
	0x2006:1	Undervoltage setpoint
	0x2006:2	Overvoltage setpoint
	0x2006:3	Overcurrent setpoint
0x2008		Cogging torque compensation
	0x2008:1	State
	0x2008:2	Enabled
0x2009		Commutation offset
	0x2009:1	State
	0x2009:2	Applied percent of rated torque

	0x2009:3	Measurement method
	0x2009:4	Phasing controller Kp
	0x2009:5	Phasing controller Ki
	0x2009:6	Phasing controller Kd
0x200A		I2t
	0x200A:1	Enabled
	0x200A:2	Peak time
	0x200A:3	Percent max
	0x200A:4	Protection active
0x200B		Max power
0x2010		Torque controller
	0x2010:1	Controller Kp (disabled)
	0x2010:2	Controller Ki (disabled)
	0x2010:3	Controller Kd (disabled)
	0x2010:4	Field weakening enabled
	0x2010:5	Field weakening percentage
	0x2010:6	Field weakening starting speed
	0x2010:7	Field weakening ending speed
	0x2010:8	Commutation angle measurement delay
	0x2010:9	Switching frequency

	0x2010:10	Settling time
	0x2010:11	Damping ratio
	0x2010:12	Encoder source
0x2011		Velocity controller
	0x2011:1	Controller Kp
	0x2011:2	Controller Ki
	0x2011:3	Controller Kd
	0x2011:4	Controller integral limit
	0x2011:5	Encoder source
0x2012		Position controller
	0x2012:1	Position loop Kp
	0x2012:2	Position loop Ki
	0x2012:3	Position loop Kd
	0x2012:4	Position loop integral limit
	0x2012:5	Velocity loop Kp
	0x2012:6	Velocity loop Ki
	0x2012:7	Velocity loop Kd
	0x2012:8	Velocity loop integral limit
	0x2012:9	Encoder source
0x2013		Gain scheduling

	0x2013:1	Enabled
	0x2013:2	Threshold velocity 0
	0x2013:3	Position loop Kp 0
	0x2013:4	Position loop Ki 0
	0x2013:5	Position loop Kd 0
	0x2013:6	Position loop integral limit 0
	0x2013:7	Velocity loop Kp 0
	0x2013:8	Velocity loop Ki 0
	0x2013:9	Velocity loop Kd 0
	0x2013:10	Velocity loop integral limit 0
	0x2013:11	Threshold velocity 1
	0x2013:12	Position loop Kp 1
	0x2013:13	Position loop Ki 1
	0x2013:14	Position loop Kd 1
	0x2013:15	Position loop integral limit 1
	0x2013:16	Velocity loop Kp 1
	0x2013:17	Velocity loop Ki 1
	0x2013:18	Velocity loop Kd 1
	0x2013:19	Velocity loop integral limit 1
0x2014		Torque window

	0x2014:1	Value
	0x2014:2	Time
0x2015		Velocity feed forward
	0x2015:1	Gain
	0x2015:2	Cutoff frequency
0x2021		Velocity feedback filter
	0x2021:1	Type
	0x2021:2	Cutoff frequency
0x2022		Position feedback filter
	0x2022:1	Type
	0x2022:2	Cutoff frequency
0x2023		Notch filter
	0x2023:1	Enabled
	0x2023:2	Center frequency
	0x2023:3	Rejection band
0x2027		Control input FIR filter
	0x2027:1	Enabled
	0x2027:2	Order
0x2030		Core temperature
	0x2030:1	Measured temperature

0x2031		Drive temperature
	0x2031:1	Measured temperature
0x2038		External scaled measurement
	0x2038:1	Scaled measurement value
	0x2038:2	Analog input
	0x2038:3	Resistance
	0x2038:4	Constant a0
	0x2038:5	Constant a1
	0x2038:6	Constant a2
	0x2038:7	Constant a3
	0x2038:8	Constant a4
	0x2038:9	Constant a5
	0x2038:10	Cutoff frequency
	0x2038:11	Upper error threshold
	0x2038:12	Lower error threshold
0x203F		Error report
	0x203F:1	Description
0x2040		Input counter
0x20E1		High resolution data
	0x20E1:1	Data 1

	0x20E1:2	Data 2
	0x20E1:3	Data 3
	0x20E1:4	Data 4
0x20F0		Timestamp
0x20F2		Assigned name
0x20F3		DC synchronization
	0x20F3:1	DC pulse time
0x2110		Encoder 1 configuration
	0x2110:1	Sensor port
	0x2110:2	Type
	0x2110:3	Resolution
	0x2110:4	Zero velocity threshold
	0x2110:5	Polarity
	0x2110:6	Singleturn offset
	0x2110:7	Access signal type
	0x2110:8	Clock frequency
	0x2110:9	Frame size
	0x2110:10	Multiturn bits
	0x2110:11	Multiturn first bit position
	0x2110:12	Singleturn bits

	0x2110:13	Singleturn first bit position
	0x2110:14	Timeout
	0x2110:15	CRC polynomial
	0x2110:16	Maximum tbusy
	0x2110:17	Status bits active value
	0x2110:18	Parity type
	0x2110:19	First clock delay
	0x2110:20	Data ordering
	0x2110:21	Endianness
	0x2110:22	Index availability
	0x2110:23	Hall sensor port
	0x2110:24	Sinewave cycles per revolution
	0x2110:25	Sinewave resolution
	0x2110:26	Sinusoidal output voltage
	0x2110:27	Filter
0x2111		Encoder 1 feedback
	0x2111:1	Raw position
	0x2111:2	Adjusted position
	0x2111:3	Velocity
0x2112		Encoder 2 configuration

	0x2112:1	Sensor port
	0x2112:2	Type
	0x2112:3	Resolution
	0x2112:4	Zero velocity threshold
	0x2112:5	Polarity
	0x2112:6	Singleturn offset
	0x2112:7	Access signal type
	0x2112:8	Clock frequency
	0x2112:9	Frame size
	0x2112:10	Multiturn bits
	0x2112:11	Multiturn first bit position
	0x2112:12	Singleturn bits
	0x2112:13	Singleturn first bit position
	0x2112:14	Timeout
	0x2112:15	CRC polynomial
	0x2112:16	Maximum tbusy
	0x2112:17	Status bits active value
	0x2112:18	Parity type
	0x2112:19	First clock delay
	0x2112:20	Data ordering

	0x2112:21	Endianness
	0x2112:22	Index availability
	0x2112:23	Hall sensor port
	0x2112:24	Sinewave cycles per revolution
	0x2112:25	Sinewave resolution
	0x2112:26	Sinusoidal output voltage
	0x2112:27	Filter
0x2113		Encoder 2 feedback
	0x2113:1	Raw position
	0x2113:2	Adjusted position
	0x2113:3	Velocity
0x2210		GPIO pin configuration
	0x2210:1	Pin 1
	0x2210:2	Pin 2
	0x2210:3	Pin 3
	0x2210:4	Pin 4
	0x2210:5	Pin 5
	0x2210:6	Pin 6
	0x2210:7	Pin 7
0x2211		GPIO output events

	0x2211:1	Timestamp setpoint
0x2212		GPIO input actions
	0x2212:1	Timestamp on rising edge
0x2213		GPIO position trigger
	0x2213:1	Position setpoint
	0x2213:2	Target encoder
	0x2213:3	Pulse width
0x2214		GPIO global options
	0x2214:1	Voltage level
0x2401		Analog input 1
0x2402		Analog input 2
0x2403		Analog input 3
0x2404		Analog input 4
0x2611		Safety digital input diagnostics
	0x2611:1	Input 1
	0x2611:2	Input 2
0x2701		Tuning command
0x2702		Tuning status
0x2703		User MOSI
0x2704		User MISO

0x2705	Setup wizard completed

Table Caption Outside Table: Profile Specific Area

Index	Name
0x603F	Error code
0x6040	Controlword
0x6041	Statusword
0x605A	Quick stop option code
0x6060	Modes of operation
0x6061	Modes of operation display
0x6062	Position demand value
0x6064	Position actual value
0x6065	Following error window
0x6066	Following error time out
0x6067	Position window
0x6068	Position window time
0x606B	Velocity demand value
0x606C	Velocity actual value
0x606D	Velocity window
0x606E	Velocity window time
0x606F	Velocity threshold

0x6070		Velocity threshold time
0x6071		Target Torque
0x6072		Max torque
0x6073		Max current
0x6074		Torque demand
0x6075		Motor rated current
0x6076		Motor rated torque
0x6077		Torque actual value
0x6079		DC link circuit voltage
0x607A		Target position
0x607B		Position range limit
	0x607B:1	Min position range limit
	0x607B:2	Max position range limit
0x607C		Home offset
0x607D		Software position limit
	0x607D:1	Min position limit
	0x607D:2	Max position limit
0x607E		Polarity
0x6080		Max motor speed
0x6081		Profile velocity

0x6083		Profile acceleration
0x6084		Profile deceleration
0x6085		Quick stop deceleration
0x6086		Motion profile type
0x6087		Torque slope
0x6088		Torque profile type
0x6091		Gear ratio
	0x6091:1	Motor revolutions
	0x6091:2	Shaft revolutions
0x6098		Homing method
0x6099		Homing speeds
	0x6099:1	Speed during search for switch
	0x6099:2	Speed during search for zero
0x609A		Homing acceleration
0x60A9		SI unit velocity
0x60B1		Velocity offset
0x60B2		Torque offset
0x60F4		Following error actual value
0x60FA		Control effort
0x60FC		Position demand internal value

0x60FD		Digital inputs
0x60FE		Digital outputs
	0x60FE:1	Physical outputs
	0x60FE:2	Bit mask
0x60FF		Target velocity
0x6502		Supported drive modes
0x6621		Safety statusword
	0x6621:1	Byte 1
	0x6621:2	Byte 2

0x1000 Device type

Identifies the type of device.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Device type	0x1000:0	UDINT	32			131-474		read-only (default)	

0x1001 Error register

This object shall provide error information. The CANopen device maps internal errors into this object. It is a part of an emergency object.

Structure of the error register

Bit	M/O	Meaning
0	M	Generic error
1	O	Current
2	O	Voltage
3	O	Temperature
4	O	Communication error (overrun, error state)
5	O	Device profile specific
6	O	reserved (always 0_b)
7	O	manufacturer-specific

If a specific error occurs the corresponding bit shall be set to 1_b. The generic error bit shall be supported. The other bits may be supported. The generic error shall be signaled at any error situation.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Error register	0x1001:0	USINT	8			0		read-only (default)	

0x1005 COB-ID SYNC

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Data			ping
CO-B-ID SY-NC	0x100-5:0	DI-NT	32			128		read-write	

0x1006 Communication cycle period

Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Com-munic-ation cycle period	0x10-06:0	DI-NT	32			100		read-write	

0x1008 Manufacturer device name

This object shall provide the name of the device as given by the manufacturer.

Name	Index-:Sub	Type	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	U-nit	Acc-ess	PDO Map-ping
Man-ufac-	0x10-08:0	STR-ING	40-0			0		read-only	

turer device name		(50)						(default)	

0x100A Manufacturer software version

This object shall provide the manufacturer software version description.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Manufacturer software version	0x100A:0	STRING (50)	40			0		read-only (default)	

0x100C Guard time

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Guard time	0x100C:0	UINT	16			0		read-write	

0x100D Life time factor

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Life time factor	0x100D:0	USINT	8			0		read-write	

0x1010 Store parameters

Provides the ability to save to permanent storage. Write the value `0x65766173` ("evas", or "save" spelled backwards in ASCII) to start the save procedure.

> **save all parameters** - this will save the current configuration to permanent storage.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Save all parameters	0x1010:1	UDINT	32			0		read-write	

0x1011 Restore default parameters

Provides the ability to load from permanent storage. Write the value `0x64616f6c` ("daol", or "load" spelled backwards in ASCII) to start the restore procedure.

> **restore all default parameters** - load the configuration from the config.csv file, stored on permanent storage

The "default" configuration is defined as the contents of the `config.csv` file.

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Restore all default parameters	0x1011:1	UDINT	32			0		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Restore communication default parameters (disabled)	0x1011:2	UDINT	32			0		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Restore application default parameters (disabled)	0x10-11:3	UDINT	32			0		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Restore manufacturer default parameters	0x10-11:4	UDINT	32			0		read-write	

0x1016 Consumer heartbeat time

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Consumer heartbeat time	0x10-16:1	DINT	32			100		read-write	

0x1017 Producer heartbeat time

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Producer heartbeat time	0x1017:0	UINT	16			1000		read-write	

0x1018 Identity object

Provides information about the device.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Vendor-ID	0x1018:1	UDINT	32			8914		read-only (default)	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Product Code	0x1018:2	UDINT	32			513		read-only (default)	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Revision number	0x1018:3	UDINT	32			16777-2162		read-only (default)	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Serial number	0x1018:4	UDINT	32			0		read-only (default)	

0x1019 Synchronous counter overflow value

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Synchronous counter overflow value	0x1019:0	USINT	8			100		read-write	

0x1023 OS command

undefined

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Command	0x1023:1	OCTET_STRING (8)	64			0		read-write	

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Status	0x1023:2	USINT	8			0		read-only (default)	

0: last command completed, no errors, no reply.
1: last command completed, no errors, reply there.
2: last command completed, error, no reply.
3: last command completed, error, reply there.
100-200: indicates how much of the command has been executed (in %, 100 = 0%, 200 = 100%).
255: command is executing (if the percentage display is not supported).

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Response	0x1023:3	OCTET_	64			0		read-only	

		STRING (8)						(default)	
Byte 0: See SubIndex 2. Byte 1: Unused. 2-n: Command specific response data.									

0x1024 OS command mode

This object shall control the OS command execution in the application specific queue. This object represents the most recent command of an application program specific queue.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
OS command mode	0x1024:0	USINT	8			0		write-only	

0x1400 Receive PDO1 parameter

Indicates the communication parameters for the RPDO 1.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used	0x1400:1	UDINT	32			2147484160		read-write	

by PDO									
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1400:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1400:3	UINT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1401 Receive PDO2 parameter

Indicates the communication parameters for the RPDO 2.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1401:1	UDINT	32			2147484416		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1401:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Inhibit time	0x14-01:3	UI-NT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1402 Receive PDO3 parameter

Indicates the communication parameters for the RPDO 3.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1402:1	UDINT	32			2147484672		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission	0x1402:2	USINT	8			1		read-write	

type									
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1402:3	UINT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1403 Receive PDO4 parameter

Indicates the communication parameters for the RPDO 4.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1403:1	UDINT	32			2147484928		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1403:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the received data become valid.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1403:3	UINT	16			0		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									

0x1600 Receive PDO1 mapping

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Con-	0x16-	UD-	32			16148-		read-	

trol-word	00:1	INT				07056		write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Mode-s of oper-ation	0x16-00:2	UD-INT	32			16169-04200		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Target Torqu-e	0x16-00:3	UD-INT	32			16180-18320		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Target pos-ition	0x16-00:4	UD-INT	32			16186-08160		read-write	
Name	Index-	Typ-	Bit	Mi-	M-	Defaul-	U-	Acce-	PDO

	:Sub	e	Si-ze	n D-at-a	ax D-at-a	t Data	nit	ss	Map-ping
Target velo-city	0x16-00:5	UD-INT	32			16273-24448		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Torqu-e off-set	0x16-00:6	UD-INT	32			16222-78160		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Tun-ing com-mand	0x16-00:7	UD-INT	32			65437-6992		read-write	

0x1601 Receive PDO2 mapping

Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-	M-ax D-	Default Data	U-nit	Acce-ss	PDO Map-ping

				at-a	at-a				
Phy-sical out-puts	0x16-01:1	UDI-NT	32			16272-59168		read-write	
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Default Data	U-nit	Acce-ss	PDO Map-ping
Bit mas-k	0x16-01:2	UDI-NT	32			16272-59424		read-write	

0x1602 Receive PDO3 mapping

Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Default Data	U-nit	Acce-ss	PDO Map-ping
Use-r MO-SI	0x16-02:1	UDI-NT	32			65450-8064		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Default Data	U-nit	Acce-ss	PDO Map-ping

Velocity offset	0x1602:2	UDINT	32			16222-12640		read-write	

0x1603 Receive PDO4 mapping

0x1800 Transmit PDO1 parameter

Indicates the communication parameters for the TPDO 1.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1800:1	UDINT	32			21474-84032		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission	0x1800:2	USINT	8			1		read-write	

type									
A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1800:3	UINT	16			1000		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1800:4	USINT	8			0		read-only (default)	
This subindex has no function and exists only for compatibility reasons.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Event timer	0x18-00:5	UI-NT	16			0		read-write	
This time (in ms) can be used to trigger an Event which handles the copying of the data and the sending of the PDO.									

0x1801 Transmit PDO2 parameter

Indicates the communication parameters for the TPDO 2.

Name	Inde-x:Su-b	Typ-e	Bi-t Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PD-O Map-ping
COB-ID used by PDO	0x18-01:1	UD-INT	32			21474-84288		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Inde-x:Su-b	Typ-e	Bi-t Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PD-O Map-ping
Trans-mis-sion type	0x18-01:2	USI-NT	8			1		read-write	

A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1801:3	UINT	16			1000		read-write	

This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1801:4	USINT	8			0		read-only (default)	

This subindex has no function and exists only for compatibility reasons.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Event	0x18-	UI-	16			0		read-	

timer	01:5	NT						write	
This time (in ms) can be used to trigger an Event which handles the copying of the data and the sending of the PDO.									

0x1802 Transmit PDO3 parameter

Indicates the communication parameters for the TPDO 3.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1802:1	UDINT	32			2147484544		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1802:2	USINT	8			1		read-write	

A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1802:3	UINT	16			1000		read-write	

This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1802:4	USINT	8			0		read-write	

This subindex has no function and exists only for compatibility reasons.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Event timer	0x1802:5	UINT	16			0		read-write	

This time (in ms) can be used to trigger an Event which handles the copying of the data and

the sending of the PDO.

0x1803 Transmit PDO4 parameter

Indicates the communication parameters for the TPDO 4.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
COB-ID used by PDO	0x1803:1	UDINT	32			2147484800		read-write	
The COB-ID is stored here. For PDO mappings 1-4, the CAN-ID is fixed depending on the node-ID and only the valid bit (bit 31) can be set in the COB-ID. A COB-ID change does not take effect until the controller or communication is restarted.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Transmission type	0x1803:2	USINT	8			1		read-write	
A number is stored in this subindex that defines the time at which the data are to be copied into the PDO message and when this is to be sent.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Inhibit time	0x1803:3	UINT	16			1000		read-write	
This subindex contains a time lock in ms. This can be used to set a time that must elapse after the sending of a PDO before the PDO is sent another time. This time only applies for asynchronous PDOs.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
reserved	0x1803:4	USINT	8			0		read-write	
This subindex has no function and exists only for compatibility reasons.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Event timer	0x1803:5	UINT	16			0		read-write	
This time (in ms) can be used to trigger an Event which handles the copying of the data and the sending of the PDO.									

0x1A00 Transmit PDO1 mapping

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Statusword	0x1A00:1	UDINT	32			16148-72592		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Modes of operation display	0x1A00:2	UDINT	32			16169-69736		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position actual value	0x1A00:3	UDINT	32			16171-66368		read-write	

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity actual value	0x1A-00:4	UDINT	32			16176-90656		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque actual value	0x1A-00:5	UDINT	32			16184-11536		read-write	

0x1A01 Transmit PDO2 mapping

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 1	0x1A-01:1	UDINT	32			60404-5328		read-write	
Na-	Index-	Typ-	Bit	Mi-	M-	Defaul-	U-	Acce-	PDO

me	:Sub	e	Si-ze	n Da-ta	ax Da-ta	t Data	nit	ss	Map-ping
Ana-log inp-ut 2	0x1A-01:2	UDI-NT	32			60411-0864		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Ana-log inp-ut 3	0x1A-01:3	UDI-NT	32			60417-6400		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Ana-log inp-ut 4	0x1A-01:4	UDI-NT	32			60424-1936		read-write	
Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Tun-	0x1A-	UDI-	32			65444-		read-	

ing stat-us	01:5	NT				2528		write	

0x1A02 Transmit PDO3 mapping

Na-me	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Default Data	U-nit	Acce-ss	PDO Map-ping
Dig-ital inp-uts	0x1A-02:1	UDI-NT	32			16271-93376		read-write	

0x1A03 Transmit PDO4 mapping

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
User MISO	0x1A-03:1	UDI-NT	32			65457-3600		read-write	
Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Defaul-t Data	U-nit	Acce-ss	PDO Map-ping
Time-stamp	0x1A-03:2	UDI-NT	32			55259-9584		read-write	

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position demand internal value	0x1A-03:3	UDINT	32			16271-27840		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity demand value	0x1A-03:4	UDINT	32			16176-25120		read-write	
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque	0x1A-03:5	UDINT	32			16182-14928		read-write	

dema-nd									

0x1C00 Sync Manager

Nam-e	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Syn-cMan 0	0x1C-00:1	USI-NT	8			1		read-only (defa-ult)	
Nam-e	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Syn-cMan 1	0x1C-00:2	USI-NT	8			2		read-only (defa-ult)	
Nam-e	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Syn-cMan 2	0x1C-00:3	USI-NT	8			3		read-only (defa-ult)	

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SyncMan 3	0x1C00:4	USINT	8			4		read-only (default)	

0x1C10 SM 0 Assignment

0x1C11 SM 1 Assignment

0x1C12 SM 2 Assignment

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 001	0x1C12:1	UINT	16			5632		read-write	
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 002	0x1C12:2	UINT	16			5633		read-write	
Name	Index:-Sub	Type	Bit Size	Min Da-	Max Da-	Default Dat-	Unit	Access	PDO Mapping

				ta	ta	a			
SubIndex 003	0x1C12:3	UINT	16			5634		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 004	0x1C12:4	UINT	16			5635		read-write	

0x1C13 SM 3 Assignment

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 001	0x1C13:1	UINT	16			6656		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 002	0x1C13:2	UINT	16			6657		read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 003	0x1C13:3	UINT	16			6658		read-write	
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
SubIndex 004	0x1C13:4	UINT	16			6659		read-write	

0x2000 Command object (disabled)

Command object was used for storing and loading of the configuration from flash memory. This object is substituted with object 0x1010 for storing and object 0x1011 for loading of the configuration. Storing of the configuration is user triggered by entering the value "save" (LittleEndian format) in the object 0x1010, subindex 1. Loading of the configuration loads the default configuration from the flash memory. It is performed during initialization of the slave by entering the value "load" (LittleEndian format) in the object 0x1011, subindex 1.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Command object (dis-	0x2000:0	UINT	16			0		read-write	

abled)									

0x2001 Commutation angle offset

In most of the cases, the position sensor is not physically mounted at the ideal angle on the body of the motor. As a result, an "offset" value should be added to the measured value of position sensor to give the correct (electrical) angle of rotor.

This object represents the required shift for position sensor output. Its range is between [0-4095] and corresponds to [0-360] electrical degrees inside the software.

Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Com-muta-tion angle offset	0x20-01:0	IN-T	16			0		read-write	

0x2002 Position control strategy

Selects the control structure used for position control:

1. Simple PID (A position PID controller directly generating the reference value for the torque controller.)



2. Cascaded PID (A position PID controller that generates the reference value for an underlying velocity controller - while that velocity controller generates the reference value for the torque controller)

image/svg+xml

τ_offset τ_ref τ Cascaded PID

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Position control strategy	0x2002:0	UINT	16	Simple PID	1		read-write	
				Cascaded PID	2			

0x2003 Motor specific settings

Please refer to your motor's data sheet for the specific values.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pole pairs	0x2003:1	USINT	8			1		read-write	
Number of magnetic pole pairs used in the motor. Please refer to your motor datasheet for this value.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque con-	0x2003:2	DINT	32			1	µNm/A	read-write	

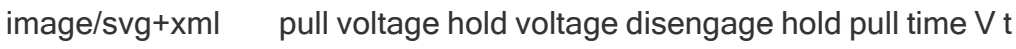
stant									
The motor's torque constant is given by the motor design it defines the torque generated per effective current(rms). The value needs to be configured in µNm/Arms									
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phase resistance	0x2003:3	DINT	32			1	µΩ	read-write	
The resistance between motor terminal and motor neutral point. It is equal to half of "phase to phase resistance" or "line to line resistance".									
Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phase inductance	0x2003:4	DINT	32			1	µH	read-write	
The inductance between motor terminal and motor neutral point. It is equal to half of "phase to phase inductance" or "line to line inductance".									
Name	Index-:Sub	Type	Bit Size	Options			Unit	Access	PDO Mapping

Motor phases inverted	0x2003:5	BOOL	1	No 0 Yes 1		read-write	
Use this parameter only when you have a negative velocity or feedback increment with the positive torque increment.							

0x2004 Brake options

Provides options for operating a brake. A brake is spring-activated such that it engages when powered off for safety. It must be disengaged by applying current to a solenoid to be released.

A disengage Pull voltage can be applied to the solenoid to disengage the brake quickly for the duration of the Pull time. After the pull time, a holding voltage is applied to keep the pin retracted.



Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pull voltage	0x2004:1	UDINT	32			0	mV	read-write	
Voltage required to disengage the brake. It is ignored in manual mode.									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Hold voltage	0x2004:2	UDINT	32			0	mV	read-write	

Voltage required to hold the brake in a disengaged state. This voltage may be set lower than the "Pull voltage" to reduce losses in the brake. CAUTION! Before setting this parameter, contact your brake manufacturer to ensure reliable operation. It is ignored in manual mode.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pull time	0x2004:3	UINT	16			0	ms	read-write	

Time period in which `0x2004:3 Pull voltage` is applied to release the brake. It is ignored in manual mode.

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Release strategy	0x2004:4	USINT	8	Manual output voltage	0		read-write	
				Clutch-style brake control	1			

				Pin brake control | 2			

Sets the method used to control the brake. The options are:

`Manual output voltage` (0) - Manually control the brake with output voltage.

`Clutch-style brake control` (1) - For brakes that will disengage even if a load is applied to them.

`Pin brake control` (2) - For brakes that require extra motion handling to ensure they release.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller disable delay	0x2004:5	UINT	16			100	ms	read-write	

Time from when the brake is commanded to engage, to when the controller is disabled. It is ignored in manual mode.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
DC bus voltage	0x2004:6	UINT	16			0	V	read-only (default)	

(deprecated)									

Deprecated. Refer to `0x6079 DC link circuit voltage` instead.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Brake status	0x2004:7	USINT	8	Not configured: 0 Engaged: 1 Disengaged: 2		read-write	

Reports the status of the brake. Can also control the brake in automatic mode (Pin brake or Clutch brake)*

The options for controlling or reading this value are:

`Not configured` (0) - No action will be taken for the brake

`Engaged` (1) - The brake will prevent motion (Output voltage disabled in manual mode)

`Disengaged` (2) - The brake will not prevent motion (Output voltage applied to 4th phase in manual mode)

* In Manual Mode, output voltage is controlled by bit 0 of `Digital Output Object (0x60FE)`

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Minimum displacement (pin brake)	0x2004:8	UDINT	32	0	2147483647	0	Inc	read-write	

Minimum angle that the motor needs to turn in before it is certain that the brake pin is pulled out. The units are in commutation sensor increments.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Percentage of Rated Current (pin brake)	0x2004:9	UINT	16	0	200	100		read-write	

Defines the amount of current to be applied during the Pin brake release. Specify this value in percent of rated current.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Output voltage	0x2004:10	UINT	16			0	mV	read-write	Receive PDO (Outputs)
Voltage which is applied on the 4th phase when the Release strategy is set to manual mode (0) and Physical output bit 0 as well as corresponding Bit mask in object `0x60FE` is set to 1.									
Name	Index:Sub	Type	Bit Size	Options			Unit	Access	PDO Mapping
Switching frequency	0x2004:11	USINT	8	16 kHz	0			read-write	
				32 kHz	1				
				64 kHz	2				
Select the switching frequency for the 4th phase (D).									

0x2006 Protection

Provides user-defined values for setting protection thresholds. If the measured value of one of these parameters exceeds the threshold, the drive will issue an error and halt the drive.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Under-	0x20-	DI-	32			0	V	read-	

voltage set-point	06:1	NT						write	
Custom value for undervoltage protection.									
Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Over-voltage set-point	0x20-06:2	DI-NT	32			100-0	V	read-write	
Custom value for overvoltage protection.									
Name	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Over-current set-point	0x20-06:3	DI-NT	32			500-000	m-A	read-write	
Custom value for overcurrent protection.									

0x2008 Cogging torque compensation

Cogging compensation is a software feature that suppresses torque ripple of electric motors to overcome vibration problems in precision applications. When cogging compensation is activated, an additional torque component is generated actively by the motor in order to compensate the cogging torque. To find the amount of compensation torque needed at each motor position, a recording procedure has to be performed before the compensation itself can be used. This object

is used for reporting the state of the motor cogging data and enabling or disabling the cogging compensation feature.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
State	0x2008:1	INT	16	NO DATA 0 IN PROGRESS 1 SAVING 2 DATA EXISTS 3		read-only (default)	
State of the cogging compensation procedure and data. Possible values: NO DATA (0) - No cogging data has been stored. Use the Cogging Recording opmode to record new data. IN PROGRESS (1) - Recording of the cogging data is in progress. SAVING (2) - Recording is done and the data is being saved to memory. DATA EXISTS (3) - Cogging data has been loaded from memory. Use the Enabled sub-field to enable the data for cogging compensation.							
Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2008:2	INT	16	Disabled 0 Enabled 1		read-write	

Enables or disables the cogging compensation feature.

0x2009 Commutation offset

Used for reporting the state of the commutation offset.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
State	0x2009:1	INT	16	OFFSET INVALID: 0 IN PROGRESS: 1 OFFSET VALID: 2		read-write	

This subitem shows the state of the commutation offset. A new offset can be found by entering the Commutation Offset Detection opmode.

`OFFSET INVALID` (0) - Offset detection needs to be executed before using the torque controller.
`IN PROGRESS` (1) - Commutation Sensor Offset Detection is currently in progress.
`OFFSET VALID` (2) - The offset has been found and is valid. The torque controller can now be used.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Applie-	0x20-	IN-	16	0	100	100		read-	

d percent of rated torque	09:2	T						write	
Defines the amount of torque to be applied during the offset detection process. Specify this value in percent of rated torque.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Measurement method	0x2009:3	SINT	8	0	2	0		read-write	

Used method for measurment of commutation angle offset
Default value is 0. This method (method-0) does not need any tuning, it might rotate the motor up to 100% of a pole-pare range during phasing procedure.
If 1 is selected, method-1 will be executed to find the proper value of commutation angle offset. Before using method-1 the following steps should be completed:

Proper value of "phase inverted" parameter should be updated in the software.
PID parameters of phasing controller should be tuned. A well tuned phasing controller will result in no visible movement during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

If 2 is selected, method-2 will be executed to find the proper value of commutation angle offset. Regarding method 2 the following points should be considered:

Method 2 does not need any movement on motor shaft
The reliability of method 2 is a function of motor design structure. As a result, for high

reliabilty applications it is always recommended to use method 0 or method 1.
If Method 2 is selected, and commutation sensor has an index, then at startup the Drive uses method 2 to find a temporary offset. Once the index is found, this temporary offset will be replaced by the offset which is saved in object dictionary.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phasing controller Kp	0x2009:4	REAL	32	0		0		read-write	

Proportional constant Kp in PID phasing controller. This controller is used to hold the position during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phasing controller Ki	0x2009:5	REAL	32	0		0		read-write	

Integral constant Ki in PID phasing controller. This controller is used to hold the position during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Phasing controller Kd	0x2009:6	REAL	32	0		0		read-write	

Derivative constant Kd in PID phasing controller. This controller is used to hold the position during offset detection procedure.
WARNING: If not tuned, the default values of "phasing controller" do not guarantee a stable situation. Before using "offset detection method 1" tune its "phasing controller" for your mechanical load. If you want to tune "phasing controller" of "offset detection method 1" make sure that your motor can freely rotate at high speeds (recommendation: start tuning by setting kp = 1, ki = 1, and kd = 0).

0x200A I2t

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Enabled	0x200A:1	USINT	8	Disabled	0		read-write	

				Enabled (current limit at 100%) 1 Enabled (error at 100%) 2					
Enables or disables the I2t protection feature.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Peak time	0x20-0A:2	UDINT	32		360-0000	300-0	ms	read-write	
This field determines how long the motor can operate while peak rms current is flowing in its windings. This time should be determined by motor manufacturer.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Percent max	0x20-0A:3	USINT	8			0		read-only (default)	
This is an index of your motor "over-heat" status. It changes between 0 and 250. At 0%, the motor is not over heated. When this value reaches 100% the motor is at its maximum allowed over heated condition, and if I2t is enabled, the maximum rms current will be limited.									

This limitation is going to be removed when the percentage goes down to 50% or less.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Protection active	0x200A:4	BOOL	1			0		read-only (default)	
As long as it is 0, i2t protection is not putting any limit on maximum rms current of your motor. Once it is 1 it means that i2t protection is limiting the "maximum rms currents" of your motor to its rated value.									

0x200B Max power

This object is a limit on maximum mechanical power on motor shaft. Wrong settings of "motor torque constant", "position sensor configurations" or "number of motor pole-pairs" can lead to misbehavior of power limitation feature. By default the power of each Drive is internally limited to 110% of its Maximum Peak Output Power value. As a result, this object will only be effective if its value is less than 110% of the drive's Maximum Peak Output Power.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max power	0x200B:0	UDINT	32	1	2147483647	2147483647	W	read-write	

0x2010 Torque controller

Range of all torque controller gains is 0 to 100,000,000

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Kp (disabled)	0x2010:1	REAL	32	0		1159479296		read-write	
In the case of using predictive controllers for current loop the unit is undefined, and the default value is 2500. Moreover, Increasing this parameter directly amplifies the amplitude of applied voltages, and as a result has an effect similar to Kp in the case of using a PID controller. Range of this torque controller gain is 0 to 100,000,000.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Ki (disabled)	0x2010:2	REAL	32	0		1193033728		read-write	
The unit of this gain is 0.16 [uV/(Second * Amp_rms)]. The default value is 40000. Range of this torque controller gain is 0 to 100,000,000.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

			ze		a				ping
Con-troller Kd (dis-abled)	0x20-10:3	RE-AL	3-2	0		0		read-write	

This parameter is undefined for predictive current controller.
Range of this torque controller gain is 0 to 100,000,000.

Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Options	U-ni-t	Acc-ess	PD-O Map-ping
Field weak-ening enable-d	0x20-10:4	UD-INT	3-2	Disabled 0 Enabled 1		read-write	

In many applications, it is required to extend the velocity range of an electric motor.

This parameter enables or disables “Field Weakening”, a feature that increases the speed range of a PM motor by weakening the magnetic field of the rotor linearly over the speed range.

The user should set 3 parameters:

1. Field Weakening Starting Speed (Subitem 6)

2. Field Weakening Ending Speed (Subitem7)

3. Percentage of Field Weakening (f) (Subitem 5)

Note:

While the field weakening feature is enabled, it is assumed that the Drive is in its normal conditions with no fault.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Field weakening percentage	0x2010:5	UDINT	32	0	100	0		read-write	

Percentage of Field weakening. This field can have an integer value in the range of [0-100], and determines the "RMS" value of phase currents which are (only) generated to weaken the field. For example, if set to 10, then 10% of rated current will be injected in motor phases to weaken the rotor field. WARNING: Based on motor type, values more than 25% can lead to instability of torque control service. While tuning, proceed with "small steps", "limited power supply voltage" and "limited power supply current".

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Field weakening starting speed	0x2010:6	UDINT	32	0	500000	0	rpm	read-write	

Field weakening is increased linearly from 0% (at starting speed point) to its final value (at ending speed point). The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Field weakening ending speed	0x2010:7	UDINT	32	0	5000000	0	rpm	read-write	

Field weakening is increased linearly from 0% (at starting speed point) to its final value (at ending speed point). The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Commutation angle measurement delay	0x2010:8	UDINT	32			0	µs	read-write	

Time taken by the sensor in reading the position value. An offset angle corresponding to this sensor measurement delay is calculated by the firmware as shown by the equation below.

$$\text{Sensor Delay Angle Offset} = \text{Electrical Rotational Speed} \times \text{Sensor Measurement Delay}$$

This angle offset value is then added into the sampled electrical angle.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Switching frequency	0x2010:9	USINT	8	16 kHz: 0 32 kHz: 1 64 kHz: 2		read-write	

Select the switching frequency for the 3 motor phases (A, B, C).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Settling time	0x2010:10	UINT	16	100	10000	2000	µs	read-write	

Settling time of current loop controllers. This value should be entered in [micro seconds]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Damping ratio	0x2010:11	UINT	16	100	10000	2000		read-write	

Damping ratio of current loop controllers. This value should be entered in [per mill]. As an

<table>
<tr><td colspan="8">example, if a damping ratio of 1.5 is required, then this value should be set to 1500.</td></tr>
<tr><td>Name</td><td>Index:Sub</td><td>Type</td><td>Bit Size</td><td>Options</td><td>Unit</td><td>Access</td><td>PDO Mapping</td></tr>
<tr><td>Encoder source</td><td>0x2010:1-2</td><td>USINT</td><td>8</td><td><table><tr><td>No encoder</td><td>0</td></tr><tr><td>Encoder 1</td><td>1</td></tr><tr><td>Encoder 2</td><td>2</td></tr></table></td><td></td><td>read-write</td><td></td></tr>
<tr><td colspan="8">Select which encoder should be used for torque control. Must be mounted to the motor shaft. Used for commutation.</td></tr>
</table>

0x2011 Velocity controller

Velocity PID Tuning (objects 2011:1...2011:4)

IMPORTANT: These values are only used in velocity control mode (not in cascaded position control)

Controller Structure

image/svg+xml

τ_{offset} τ_{con} τ_{ref} τ Velocity Control

PID Parameter Range

The constants K_p, K_i, and K_d should be between 0 and 10 with the precision of 6-digit floating point. To find the proper values of PID constants of velocity controller the following steps are recommended:

Recommended Tuning Steps

- Step 1. Check if all constants K_p, K_i, and K_d are set to 0. After that activate the velocity controller. By default, the integral limit of velocity controller should be set to the maximum torque of selected motor in [mNm].
- Step 2. Set the reference velocity to 50% of rated motor velocity [rpm].
- Step 3. Increase K_p until the measured speed is around 70% of the reference speed. As a suggestion start with K_p equal to 0.1, and in each step increase K_p with a factor of 2.
- Step 4. From this step forth, step response of the system should be checked. Increase K_i step by step to increase the speed of velocity controller (as a suggestion start with K_i equal to 0.01 and in each step increase it by a factor of 2). At this point, the parameters K_p and K_i of the velocity controller are tuned. The parameter K_d of the velocity controller can be kept at its default value (0).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Kp	0x2011:1	REAL	32	0	65535	0		read-write	
The P-gain of the PID Velocity Control Loop [mNm/rpm].									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controller Ki	0x2011:2	REAL	32	0	65535	0		read-write	
The I-gain of the PID Velocity Control Loop [mNm/(rpm*s)]									
Name	Index:Sub	Type	Bit Si-	Min Dat-	Max	Default	Unit	Access	PDO Map-

			ze	a	Dat-a	Dat-a			ping
Con-troller Kd	0x20-11:3	RE-AL	32	0	655-35	0		read-write	
The D-gain of the PID Velocity Control Loop [rpm*s/inc]									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Min Dat-a	Ma-x Dat-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Con-troller integ-ral limit	0x20-11:4	UDI-NT	32	0		0	m-N-m	read-write	
Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually.									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Options			Un-it	Acce-ss	PDO Map-ping
Enco-der sour-ce	0x20-11:5	USI-NT	8	No encoder	0			read-write	
				Encoder 1	1				

				Encoder 2 | 2			
Select which encoder should be used for velocity control.							

0x2012 Position controller

Parameter Tuning Guide for Position Controller with Cascaded Structure

Introduction

The cascaded control structure is the standard position control structure for SOMANET Drives. Its controller gains can be automatically set by the auto-tuning process, the results will be stored in the objects 2012:1 ... 2012:8. If auto-tuning is not applicable or manual modifications are necessary, the gains can be set manually. In the following, a procedure to do so is described. If a simple PID position controller is selected instead of the cascaded PID position controller, its gains are defined by the objects 2012:1 ... 2012:4.

Control Basics

image/svg+xml

τ_{offset} τ_{ref} τ Cascaded PID

Control Basics

Cascaded controller is usually used in PI-P form, i.e. inner loop (velocity control loop) is only using the proportional part of its PID controller and the outer loop (position controlling loop) is only using its proportional and integral part. The other form is P-PI form, i.e. inner loop is using proportional and integral part and outer loop uses only proportional part. The inner velocity control loop is responsible for calculating the torque reference. As a result the integral limit of the velocity controller should be set to the maximum torque of your torque actuato. Moreover, the outer position control loop is responsible for controlling the reference velocity, and consequently its integral limit should be set to the maximum velocity of your system.

Tuning Concept in Brief

For tuning the cascaded structure, we should first focus on the inner (velocity) loop. As it is the inner controlling loop, it should be faster than the outer position loop. However, if all parameters of the outer position controller are set to 0, the inner loop will be disconnected from the user position commands. As a result, we can increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity` = 10 * K_p_position`). Once the real position started to follow the reference position, we can stop increasing the K_p_position, and only focus on increasing K_p_velocity. At this stage, we can increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is caused by controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise. Once the velocity controller (the inner controlling loop) is tuned, it is time to tune the position controller (the outer controlling loop). To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of too sharpened position control. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. So far, the proportional parts of both inner velocity loop and outer position loop are tuned, and we can focus on integrator part of the outer position or inner velocity loop (depends on chosen structure of controller). Increasing the integrator constant will remove the steady state error, but at the same time, it adds some overshoot at step responses. As a rule of thumb, you can increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

In the following section, the explained tuning concept is divided in separate systematic steps.

Tuning Steps

- Step 1. Set the PID constants of both controllers equal to 0. By default, the integral limit of the velocity controller should be set to motor maximum torque and the integral limit of position controller should be set to motor maximum velocity. **From this step forth, the step response of position controller should always be evaluated.**
- Step 2. In this step, the K_p constant of the velocity controller should be tuned. Increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity = 10 * K_p_position). Once the real position starts to follow the reference position, stop increasing the K_p_

position and only focus on increasing the K_p_velocity. Increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is because of controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise.

- Step 3. Now that the velocity controller is tuned, it is time to tune the parameters of the PID position controller. Start with K_p_position, and increase it until you again get close to the instability margin, and then reduce K_p_position` to its 90%. So far, the proportional parts of both inner velocity loop and outer position loop are tuned.
- Step 4. In this step, increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Parameter Tuning Guide for Position Controller with simple PID Structure (objects 2012:1...2012:4 only)

The Position Controller with Cascaded Structure is generally preferable over the simple PID controller. Apart from academic purposes, there are few reasons to use the simple PID controller.

Control Structure

image/svg+xml

τ_offset τ_ref τ Simple PID

Tuning Concept for Simple PID Controller

To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of a position control that has been sharpened too much. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. Now increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kp	0x2012:1	REAL	32	0	65535	0		read-write	
The P-gain of the PID Position Control Loop. The value is given in [rpm/inc] for the cascaded position controller ([mNm/inc] for the simple PID position controller)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Ki	0x2012:2	REAL	32	0	65535	0		read-write	
The I-gain of the PID Position Control Loop. The value is given in [rpm/(inc*s)] for the cascaded position controller ([mNm/(inc*s)] for the simple PID position controller)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Pos-	0x20-	RE-	32	0	655-	0		read-	

ition loop Kd	12:3	AL			35			write	
The D-gain of the Position Control Loop [rpm*s/(inc)] for the cascaded position controller ([mNm*s/(inc)] for the simple PID position controller)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop integral limit	0x2012:4	UDINT	32	0		0	rpm	read-write	
Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kp	0x2012:5	REAL	32	0	65535	0		read-write	

The P-gain of the PID Velocity Loop Controller [mNm/rpm]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Ki	0x2012:6	REAL	32	0	65535	0		read-write	
The I-gain of the PID Velocity Controller [mNm/(rpm*s)]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kd	0x2012:7	REAL	32	0	65535	0		read-write	
The D-gain of the PID Velocity Loop Controller [mNm*s/rpm]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop integral	0x2012:8	UDINT	32	0		0	mNm	read-write	

limit									
Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually.									
Name	Index:Sub	Type	Bit Size	Options			Unit	Access	PDO Mapping
Encoder source	0x2012:9	USINT	8	No encoder	0			read-write	
				Encoder 1	1				
				Encoder 2	2				
Select which encoder should be used for position control.									

0x2012 Position controller

Parameter Tuning Guide for Position Controller with Cascaded Structure

Introduction

The cascaded control structure is the standard position control structure for SOMANET Drives. Its controller gains can be automatically set by the auto-tuning process, the results will be stored in the objects 2012:1 ... 2012:8. If auto-tuning is not applicable or manual modifications are necessary, the gains can be set manually. In the following, a procedure to do so is described. If a simple PID position controller is selected instead of the cascaded PID position controller, its gains are defined by the objects 2012:1 ... 2012:4.

Control Basics

image/svg+xml

τ_offset τ_ref τ Cascaded PID

Control Basics

Cascaded controller is usually used in PI-P form, i.e. inner loop (velocity control loop) is only using the proportional part of its PID controller and the outer loop (position controlling loop) is only using its proportional and integral part. The other form is P-PI form, i.e. inner loop is using proportional and integral part and outer loop uses only proportional part. The inner velocity control loop is responsible for calculating the torque reference. As a result the integral limit of the velocity controller should be set to the maximum torque of your torque actuato. Moreover, the outer position control loop is responsible for controlling the reference velocity, and consequently its integral limit should be set to the maximum velocity of your system.

Tuning Concept in Brief

For tuning the cascaded structure, we should first focus on the inner (velocity) loop. As it is the inner controlling loop, it should be faster than the outer position loop. However, if all parameters of the outer position controller are set to 0, the inner loop will be disconnected from the user position commands. As a result, we can increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity` = 10 * K_p_position`). Once the real position started to follow the reference position, we can stop increasing the K_p_position, and only focus on increasing K_p_velocity. At this stage, we can increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is caused by controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise. Once the velocity controller (the inner controlling loop) is tuned, it is time to tune the position controller (the outer controlling loop). To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of too sharpened position control. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. So far, the proportional parts of both inner velocity loop and outer position loop are tuned, and we can focus on integrator part of the outer position or inner velocity loop (depends on chosen structure of controller). Increasing the integrator constant will remove the steady state error, but at the same time, it adds some overshoot at step responses. As a rule of thumb, you can increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

In the following section, the explained tuning concept is divided in separate systematic steps.

Tuning Steps

- Step 1. Set the PID constants of both controllers equal to 0. By default, the integral limit of the velocity controller should be set to motor maximum torque and the integral limit of position controller should be set to motor maximum velocity. **From this step forth, the step response of position controller should always be evaluated.**
- Step 2. In this step, the K_p constant of the velocity controller should be tuned. Increase the K_p of both position and velocity controllers with a ratio of 10 (K_p_velocity = 10 * K_p_position). Once the real position starts to follow the reference position, stop increasing the K_p_position and only focus on increasing the K_p_velocity. Increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase K_p_velocity until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is because of controller sharpness). Now, you can reduce K_p_velocity to 90% of its value to increase the stability margin, and remove vibration noise.
- Step 3. Now that the velocity controller is tuned, it is time to tune the parameters of the PID position controller. Start with K_p_position, and increase it until you again get close to the instability margin, and then reduce K_p_position` to its 90%. So far, the proportional parts of both inner velocity loop and outer position loop are tuned.
- Step 4. In this step, increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Parameter Tuning Guide for Position Controller with simple PID Structure (objects 2012:1...2012:4 only)

The Position Controller with Cascaded Structure is generally preferable over the simple PID controller. Apart from academic purposes, there are few reasons to use the simple PID controller.

Control Structure

image/svg+xml τ_offset τ_ref τ Simple PID

Tuning Concept for Simple PID Controller

To tune the position controller loop, we should start with the P part of its PID controller. Increase K_p_position until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of a position control that has been sharpened too much. At this step, reduce K_p_position to its 90% to increase the stability margin and remove the vibration/acoustic noise. Now increase K_i to eliminate the steady state error. As a suggestion start with K_i equal to 0.01, and in each step, increase it with a factor of 2. Increase K_i step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kp	0x2012:1	REAL	32	0	65535	0		read-write	
The P-gain of the PID Position Control Loop. The value is given in [rpm/inc] for the cascaded position controller ([mNm/inc] for the simple PID position controller)									

Name	Index:Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Dat-a	Dat-a			ping
Pos-ition loop Ki	0x20-12:2	RE-AL	32	0	655-35	0		read-write	
The I-gain of the PID Position Control Loop. The value is given in [rpm/(inc*s)] for the cas-caded position controller ([mNm/(inc*s)] for the simple PID position controller)									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Dat-a	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Pos-ition loop Kd	0x20-12:3	RE-AL	32	0	655-35	0		read-write	
The D-gain of the Position Control Loop [rpm*s/(inc)] for the cascaded position controller ([mNm*s/(inc)] for the simple PID position controller)									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Dat-a	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Pos-ition loop integ-ral limit	0x20-12:4	UDI-NT	32	0		0	rp-m	read-write	

Integrator limitation for anti-windup. Technique used is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Initially set to MAX (2,147,483,647 --> Integral limit DEACTIVATED), please set the maximal values for your integral limit manually. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

0x2013 Gain scheduling

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2013:1	BOOL	1	Disabled: 0 Enabled: 1		read-write	
Enables or disables the gain scheduling feature.							

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Threshold velocity 0	0x2013:2	UDINT	32	0		0	rpm	read-write	
Velocity that determines the lower threshold. Underneath this value, controller gains set 0 is used. Beyond it, linear interpolation begins. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).									

Name	Index:Sub	Type	Bit Si-	Min Dat-	Max	Default	Unit	Access	PDO Map-

			ze	a	Dat-a	Dat-a			ping
Pos-ition loop Kp 0	0x20-13:3	RE-AL	32	0	655-35	0		read-write	
The P-gain of the Position Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm/inc]									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Min Dat-a	Ma-x Dat-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Pos-ition loop Ki 0	0x20-13:4	RE-AL	32	0	655-35	0		read-write	
The I-gain of the Position Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm/(inc*s)]									
Nam-e	Index-:Sub	Typ-e	Bit Si-ze	Min Dat-a	Ma-x Dat-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Pos-ition loop Kd 0	0x20-13:5	RE-AL	32	0	655-35	0		read-write	
The D-gain of the Position Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm*s/inc]									

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop integral limit 0	0x2013:6	UDINT	32	0		0	rpm	read-write	
The integrator limit of the Position Control Loop of area 0 (low velocity, below threshold velocity 0). Technique used for preventing the windup is clamping and it prevents the integral term from accumulating above or below pre-determined bounds. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kp 0	0x2013:7	REAL	32	0	65535	0		read-write	
The P-gain of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0) [mNm/rpm]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

					a	a			
Velocity loop Ki 0	0x2013:8	REAL	32	0	65535	0		read-write	
The I-gain of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0) [mNm/(rpm*s)]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kd 0	0x2013:9	REAL	32	0	65535	0		read-write	
The D-gain of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0) [rpm*s/inc]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop integral limit 0	0x2013:10	UDINT	32	0		0	mNm	read-write	

The integrator limit of the Velocity Control Loop of area 0 (low velocity, below threshold velocity 0). Technique used for preventing the windup is clamping and it prevents the integral term from accumulating above or below pre-determined bounds.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Threshold velocity 1	0x2013:11	UDINT	32	0		0	rpm	read-write	

Velocity that determines the upper threshold. Above this value, controller gains set 1 is used. Up to this velocity linear interpolation is performed. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kp 1	0x2013:12	REAL	32	0	65535	0		read-write	

The P-gain of the Position Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm/inc]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Position loop Ki 1	0x2013:13	REAL	32	0	65535	0		read-write	

The I-gain of the Position Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm/(inc*s)]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop Kd 1	0x2013:14	REAL	32	0	65535	0		read-write	

The D-gain of the Position Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm*s/inc]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position loop integral limit 1	0x2013:15	UDINT	32	0		0	rpm	read-write	

The integrator limit of the Position Control Loop of area 1 (high velocity, above threshold velocity 1). Technique used for preventing the windup is clamping and it prevents the integ-

ral term from accumulating above or below pre-determined bounds. Limit should always be set in the driving shaft units, especially important, when the gear box is used in the setup. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Kp 1	0x2013:16	REAL	32	0	65535	0		read-write	

The P-gain of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1) [mNm/rpm]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop Ki 1	0x2013:17	REAL	32	0	65535	0		read-write	

The I-gain of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1) [mNm/(rpm*s)]

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Velocity loop Kd 1	0x20-13:18	REAL	32	0	655-35	0		read-write	
The D-gain of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1) [rpm*s/inc]									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity loop integral limit 1	0x20-13:19	UDINT	32	0		0	m-N-m	read-write	
The integrator limit of the Velocity Control Loop of area 1 (high velocity, above threshold velocity 1). Technique used for preventing the windup is clamping and it prevents the integral term from accumulating above or below pre-determined bounds.									

0x2014 Torque window

undefined

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Value	0x2014:1	UINT	16			65535		read-write	
This object shall indicate the configured symmetrical range of accepted torque values rel-									

ative to the target torque. If the actual torque is within the torque window, target torque shall be regarded as having been reached. The value shall be given per thousand of rated torque. If the value of the torque window is 0xFFFF, torque window monitoring will be switched off.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Time	0x2014:2	UINT	16			0	ms	read-write	
This object shall indicate the configured time, during which the actual torque within the torque window is measured. The value shall be given in ms.									

0x2015 Velocity feed forward

Velocity feed-forward takes the derivative of the target position and filters the result by using the 1st order low pass filter. The output of the filter is then multiplied by the configured gain.

The end result is added as an additional input to the velocity controller in cascaded position control mode.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Gain	0x2015:1	INT	16	0	5000	0		read-write	
Setting the gain to 0 disables the feedforward feature. Any other value configures the end gain and enables the feature. The value is given in per-mil percent of estimated velocity.									

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Cutoff frequency	0x2015:2	UINT	16	5	2000	20	Hz	read-write	
Cutoff frequency of the 1st order low pass filter that filters the calculated velocity.									

0x2021 Velocity feedback filter

Measured velocity signal from the sensor can be filtered. There is only one type of filtering that the user can choose:
1st order low pass filter with formula G(s)=1/(1+sT), where T=1/(2*pi*f),
The filter is implemented in discrete domain using Euler's forward difference discretization method, i.e. s=1/T*(1-z^-1).

The filter is enabled by setting the filter type parameter to 1, and disabled by setting it to 0. Parameter cutoff frequency is set in [Hz].

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Type	0x2021:1	USINT	8	Disabled	0		read-write	
				1st order low pass	1			

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Cutoff frequency	0x2021:2	UDINT	32			0	Hz	read-write	

0x2022 Position feedback filter

Measured position signal from the sensor can be filtered. There is only one type of filtering that the user can choose:
1st order low pass filter with formula G(s)=1/(1+sT), where T=1/(2*pi*f),
The filter is implemented in discrete domain using Euler's forward difference discretization method, i.e. s=1/T*(1-z^-1).

The filter is enabled by setting the filter type parameter to 1, and disabled by setting it to 0. Parameter cutoff frequency is set in [Hz].

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Type	0x2022:1	USINT	8	Disabled: 0 1st order low pass: 1		read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Cutoff frequency	0x2022:2	UDINT	32			0	Hz	read-write	

0x2023 Notch filter

Implemented as IIR 2nd order filter configured by the center frequency and the bandwidth around it. Filter is placed on the output of the velocity controller so it filters the torque demand value. Frequency prewarping is applied. Filter formula in discrete domain

$y_k = a_0 x_k + a_1 x_{k-1} + a_2 x_{k-2} + b_1 y_{k-1} + b_2 y_{k-2}$

where $a_0 = K$, $a_1 = -2K\cos(2\pi f)$, $a2 = K$, $b1 = 2R\cos(2\pi f)$, $b2 = -R^2$

and K = (1 - 2Rcos(2pi*f) + R^2) / (2-2cos(2pi*f))

R = 1 - 3 * BW

f - center frequency and BW - bandwidth expressed relative to the sampling frequency, where f,BW = [0, 0.5]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2023:1	BOOL	1	Disabled 0 Enabled 1		read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Center frequency	0x2023:2	UINT	16	10	2000	100	Hz	read-write	

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Rejection band	0x2023:3	UINT	16	1	4000	1	Hz	read-write	

0x2027 Control input FIR filter

FIR filter (boxcar)

Depending on the mode of operation, this filter acts on the position or velocity demand values sent from the master. The sampling frequency is the same as the cycle time of the position controller (4 kHz). It can be used as an input profiler (or "shaper") to realize linear interpolation between successive commands (see documentation), or to limit the jerk of the profilers (see documentation). The default value is suited for masters operating at 1 kHz.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Enabled	0x2027:1	BOOL	1	Disabled 0 Enabled 1		read-write	
Set to 1 to enable the control input FIR filter.							

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Order	0x2027:2	USINT	8	0	31	3		read-write	
The order or length of the FIR filter. Order of the filter means the maximum number of delay elements used in the filter circuit.									

0x2030 Core temperature

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Measured temper-	0x2030:1	DINT	32			0	m°C	read-only (default)	Transmit PDO (Inpu-

ature									ts)
Describes the temperature measured on the Core board.									

0x2031 Drive temperature

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Measured temperature	0x2031:1	DINT	32			0	m°C	read-only (default)	Transmit PDO (Inputs)
Describes the temperature measured near the inverter.									

0x2038 External scaled measurement

Feature that takes an analog input and transforms it to an output as defined by a 5th order polynomial.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Scaled measurement value	0x2038:1	REAL	32			0	1	read-only (default)	Transmit PDO (Inputs)

Scaled value of the selected analog input after applying the selected 5th order polynomial.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Analog input	0x2038:2	USINT	8	Internal: 0 Analog input 1: 1 Analog input 2: 2 Analog input 3: 3 Analog input 4: 4		read-write	

Select the scaled measurement source.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Resistance	0x2038:3	UDINT	32			0	Ω	read-write	

Select the pull-up resistor connected to the selected analog input.
In case there is no pull-up resistor "0" should be selected. In that case, the polynomial would be applied to the ADC count directly (range: 0 - 4095).
Otherwise, it will be assume the user connected a pull-down resistor to create a voltage divider. The pull-down resistor value will be calculated and the polynomial will be applied to

that value.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Constant a0	0x2038:4	REAL	32			0		read-write	
Constant a0 of the polynomial (Offset)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Constant a1	0x2038:5	REAL	32			0		read-write	
Constant a1 of the polynomial (1st order coefficient)									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Constant a2	0x2038:6	REAL	32			0		read-write	
Constant a2 of the polynomial (2nd order coefficient)									
Name	Index:Sub	Type	Bit	Min	Max	Default Data	Unit	Access	PDO

	b		Si-ze	Da-ta	Da-ta		t		Map-ping
Con-stant a3	0x20-38:7	RE-AL	3-2			0		read-write	
Constant a3 of the polynomial (3rd order coefficient)									
Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Mi-n Da-ta	Ma-x Da-ta	Defau-lt Data	U-ni-t	Acc-ess	PD-O Map-ping
Con-stant a4	0x20-38:8	RE-AL	3-2			0		read-write	
Constant a4 of the polynomial (4th order coefficient)									
Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Mi-n Da-ta	Ma-x Da-ta	Defau-lt Data	U-ni-t	Acc-ess	PD-O Map-ping
Con-stant a5	0x20-38:9	RE-AL	3-2			0		read-write	
Constant a5 of the polynomial (5th order coefficient)									
Name	Inde-x:Su-b	Ty-pe	Bi-t Si-ze	Mi-n Da-ta	Ma-x Da-ta	Defau-lt Data	U-ni-t	Acc-ess	PD-O Map-ping

Cutoff frequency	0x2038:10	UINT	16	1	1000	500	Hz	read-write	
Cutoff frequency (Hz) for the LP filter for external scaled analog measurements.									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Upper error threshold	0x2038:11	REAL	32			1199570688		read-write	
Upper error threshold for the external scaled measurements. When the scaled measurement value exceeds this threshold, a fault is generated. It is also used to generate a warning when the scaled measurement value is within 5% of the error threshold. The default value corresponds to the equivalent int representation of the float value 65535.0									
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Lower error threshold	0x2038:12	REAL	32			0		read-write	
Lower error threshold for the external scaled measurements. When the scaled measurement value goes lower than this threshold, a fault is generated. It is also used to generate a warning when the scaled measurement value is within 5% of the error threshold.									

0x203F Error report

Provides further information about the error or warning reported by the `0x603F Error code` object.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Description	0x203F:1	STRING (8)	64			0		read-only (default)	Transmit or Receive PDO

The error report object display information about a fault or warning that has occured.
When a fault on the drive is reported, the drive switches into Fault reaction active state, which issues a quick-stop, brings the drive to a halt, and then disables the power stage. Thus, only one error is reported at a time.
When a warning is reported, the state of the drive remains unchanged. A subsequent fault overwrites an existing warning.

Error Code	Description	Explanation	Remedy
NoFault	No fault	Default value of the object, which means no fault is to be reported.	This error code indicates that there's no error reported, thus the drive is not in fault state. If the drive is shown in fault state and this error

Error Code	Description	Explanation	Remedy
			code is reported, this may be caused by a fault in the master/slave communication for example in the VM version OBLAC Drives. Rebooting the VM might solve the problem in this case. Another reason could be that the fault register could not be read. Otherwise please contact technical support of Synapticon.
Protection faults of user-defined parameters		Errors triggered by user-defined thresholds (for more details see object 0x2006: Protection)	NOTE: These errors can also cause subsequent faults or be caused by other faults.
PuOcA	Protection User-defined Over-cur-	The current	As the threshold is

Error Code	Description	Explanation	Remedy
	rent Phase A	sensor at drive phase A/B/C sensed a current exceeding the user-defined Over-current threshold (0x2006:3 Protection Over-current setpoint). If the user-defined threshold is higher than the hardware protection threshold, this software protection can be considered unfunctional, as long as the hardware protection is available.	user-defined, please first check the object 0x2006:3 Over-current setpoint to make sure that the threshold is set appropriately. Regarding an over-current, please refer to the remedy for over-current.
PuOcB	Protection User-defined Over-current Phase B		
PuOcC	Protection User-defined Over-current Phase C		
PuUv	Protection User-defined Under-voltage	Indicates that the DC bus voltage is measured below the user-defined threshold (0x2006:1 Protection Under-voltage setpoint).	As the threshold is user-defined, please first check the object 0x2006:1 Under-voltage setpoint. Then please refer to the remedy for

Error Code	Description	Explanation	Remedy
			under-voltage.
PuOv	Protection User-defined Over-voltage	Indicates that the DC bus voltage is measured exceeding the user-defined threshold (0x2006:2 Protection Over-voltage setpoint).	As the threshold is user-defined, please first check the object 0x2006:2 Over-voltage setpoint. Then please refer to the remedy for over-voltage.
Hardware Protection Faults		Errors triggered by hardware built-in protection logic.	
PhUdef	Protection Hardware Undefined	Indicates that the hardware protection reports an error not defined in the CiA 402 standard.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
PhFault	Protection Hardware Fault	A generic hardware fault was detected.	Please note down the information regarding the

Error Code	Description	Explanation	Remedy
			occurence of the error and reach Synapticon technical support for further investigation.
PhWtdg	Protection Hardware Watchdog	The watchdog is a service that checks periodically with a timer to guarantee that the drive control logic is working in the proper frequency. This error indicates that the watchdog tick is not received on time by the hardware protection, thus the drive is not working properly.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
PhDeadTA	Protection Hardware Deadtime Phase A	Deadtime is a common protection strategy in PWM control for avoiding a bridge	Please note down the information regarding the occurence of the error and reach

Error Code	Description	Explanation	Remedy
		shoot-through, a short-circuit accident caused by transistors on both sides being open at the same time. This error indicates that the effective deadtime is detected below the manufacturer-defined threshold.	Synapticon technical support for further investigation.
PhDeadTB	Protection Hardware Deadtime Phase B		
PhDeadTC	Protection Hardware Deadtime Phase C		
PhDeadTD	Protection Hardware Deadtime Phase D		
PhOvUvOt	Protection Hardware Overvoltage / Undervoltage / Overtemperature	The drive temperature which was measured close to the power MOSFETs and the DC bus voltage are checked by the same hardware protection circuit against manufacturer-defined thresholds. This error indicates that one or more of the three errors is/are triggered.	Please refer to the remedy for overvoltage, undervoltage and overtemperature.

Error Code	Description	Explanation	Remedy
PhOc	Protection Hardware Overcurrent	Indicate that the phase current or the DC bus current measured is exceeding the manufacturer-defined threshold (referring to the hardware document, link), which triggers the hardware protection circuit.	Please refer to the remedy for overcurrent.
Safety Faults			
SfeDiIvd	Safety Digital Input Invalid	Invalid digital input signals received by the safety input, for example one signal of the dual-channel input is read at a different interval than the other signal. The signals must have the same state within 100	The cause of the fault usually directs to misuse of the I/Os or faults in signal tansfer. Please refer to the documentation about safety functions of your hardware first, to check the use of the safety I/Os and of

Error Code	Description	Explanation	Remedy
		ms.	the safety functions. Check the measured inputs in 0x2611 Safety digital input diagnostics and locate the signal leading to the fault. Check and fix the wiring accordingly with assistance of tools for example with multimeter measurement.
SfeFault	Safety Fault	A safety-based hardware fault detected.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
BiSS Sensor Faults			
BisErBit	BiSS Error Bit active.	One of the bits in the BiSS frame is	Please check the 2

Error Code	Description	Explanation	Remedy
		used to signal error. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the error bit being active.	cases often seen: 1. Misconfiguration of Status bits active value. The status bits (error bit and warning bit) are by default read as active low, i.e. the error bit being 0 indicates the error being active. Please check the documentation of the encoder and correctly configure the reading. 2. The encoder is indeed rasing an error/ warning. Please check the documentation of the encoder, consult the encoder manufacturer for further investigation.
BisWnBit	BiSS Warning Bit active.	One of the bits in the BiSS frame is used to signal warning. This bit is set by the encoder and read by the drive according to user-defined parameter (Status bits active value). This fault indicate that the drive receives the warning bit being active.	

Error Code	Description	Explanation	Remedy
BisAcBit	BiSS Acknowledge Bit not received.	The BiSS slave is not acknowledging the request to communicate sent by the drive. This usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	Please firstly make sure that the encoder is connected to the port which is being configured. Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
BisSrtBt	BiSS Start Bit not received.	The BiSS slave acknowledged the request to communicate, but it didn't sent the start bit to start the communication. This usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or	

Error Code	Description	Explanation	Remedy
		not well connected.	
BisSloLo	BiSS data (SLO) is permanently Low.	The SLO line should be set to high by the BiSS slave when it is ready to establish communication. This error usually happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	
BisFrame	BiSS Frame configuration error.	The sensor configuration causes an issue related to the BiSS frame. This error is currently raised only if the BiSS frame size is higher than 64 bits.	Please check the sensor configuration according to the documentation of the encoder manufacturer. The most relevant objects to check: 1. Resolution, unit

Error Code	Description	Explanation	Remedy
			Inc/ Revolution 2. Multiturn resolution, unit number of bits 3. Number of filling bits, unit number of bits
BisCrc	BiSS CRC error.	The BiSS frame contains a CRC6 of some of the data that is being sent. This error is raised when the CRC sent by the BiSS slave is different than the CRC computed by the drive. This usually means that the drive can't read the frame properly.	Regarding a CRC error, there are several causes often seen: 1. Misconfiguration of the sensor reading leading to the bits being read wrong. In this case the fault cannot be reset. Please check the sensor configuration according to the documentation of the encoder manufacturer. Common parameters wrongly configured are the singleturn

Error Code	Description	Explanation	Remedy
			resolution and multiturn resolution. 2. Misconfiguration of the object CRC polynom, thus the CRC is calculated wrong. In this case the fault cannot be reset. Please check the polynom in the encoder documentation. The value by default: 48 represents x^0 + x^1 + x^6, which usually doesn't need to be modified. 3. Inconsistency in signal transfer. In this case the fault can sometimes be reset, but would occur under certain circumstances. Please check the signals received by the drive with

Error Code	Description	Explanation	Remedy
			an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
BisRegEr	BiSS register communication error	Some BiSS encoders require the firmware to read/write some of their registers to	Try to reconfigure the encoder that raised this error. To do so, there are two possibilities: -

Error Code	Description	Explanation	Remedy
		be initialized properly. This error is triggered when any of these BiSS register transaction fail.	Set its function to Disabled and then back to the function it had. - Remove the encoder from the sensor port list (0x2100) and add it again.
SSI Sensor Faults			
SsiFrame	SSI Frame configuration error.	The sensor configuration causes an issue related to the SSI frame. This error is currently raised only if the SSI frame size is higher than 256 bits.	Please check the sensor configuration, especially the object Frame size according to the documentation of the encoder manufacturer.
SsiDtaLo	SSI Data is permanently Low.	The data line should be set to high by the SSI slave when it is ready to establish communication. This error usually	Please firstly make sure that the encoder is connected to the port which is being configured.

Error Code	Description	Explanation	Remedy
		happens when there is some issue with the power or communication link, e.g. one of the wires is broken or not well connected.	Then check the wiring accoding to the hardware wiring instructions to make sure that, firstly the signals are transfered to the correct pins, secondly the signal transfer is stable. Assistance of measurement tools such as multimeter and oscilloscope is necessary.
SsiChksm	SSI Checksum error.	Some SSI slaves include a ckeck-sum of the data in the SSI frame. Currently our drives can be configured to read a parity bit from compatible SSI encoders. This error means the parity bit com-	Regarding a check-sum error, there are several causes often seen: 1. Misconfiguration of the sensor leading to the bits being read wrong or the checksum being calculated

Error Code	Description	Explanation	Remedy
		puted by the firmware is different than the one sent by the SSI slave.	wrong. In this case the fault cannot be reset. Please check the sensor configuration according to the documentation of the encoder manufacturer. 2. Inconsistency in signal transfer. In this case the fault can sometimes be reset, but would occur under certain circumstances. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is

Error Code	Description	Explanation	Remedy
			unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
REM16MT Sensor Faults			
R16WkMgF	REM16 Weak Magnetic Field.	Weak magnetic field. Mounting may need to be improved.	
R16Cont	REM16 multiturn Counter error.	Multiturn counter error	
R16CorDc	REM16 singleturn CORDIC error	Singleturn CORDIC error	

Error Code	Description	Explanation	Remedy
R16SpdOv	REM16 multiturn Speed Overflow.	Multiturn speed overflow	
R16FtCfg	REM16 Filter Configuration error.	Filter configuration error	
R16FtSOF	REM16 Filter Speed Overflow error.	Filter speed overflow error	
R16UnCmd	REM16 Unknown Command.	Unknown command	
R16Chksm	REM16 Checksum error.	Checksum error	
REM14 Sensor Faults			
R14Chksm	REM14 Checksum error.	Checksum error	
QEI Sensor Faults			
QeiLsTck	QEI Lost Ticks.	The number of ticks counted in two successive index pulses is different from the sensor resolution	There are several causes often seen: 1. The configured resolution is

Error Code	Description	Explanation	Remedy
		configured by user.	wrong. Please check the Resolution configured according to the encoder datasheet, and please pay attention to the conversion between units (Resolution = Cycles per Revolution * 4). 2. There's a loss of pulses in encoder reading, which can be due to: 2a. The encoder being improperly mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and consult the encoder manufacturer for

Error Code	Description	Explanation	Remedy
			advices. 2b. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable

Error Code	Description	Explanation	Remedy
			noise. 2c. The pulses are read with an exceeding frequency. This case rarely happens, but can be possible if the encoder resolution and/or the spin speed being unusually high.
QeiNoIdx	QEI No Index.	For an incremental encoder configured to be with 3 channels (usually known as ABI/ ABZ), the drive is expecting to receive an Index pulse which provides an index for the absolute singleturn position. This error indicates that the Index pulse couldn't be detected, while	Please first check whether your encoder is indeed with the Index pulse, i.e. is an ABI/ ABZ encoder. If it is an encoder without index, please choose 2-channel (no index) for Number of channels. Else, most probable causes are: 1. The encoder being improperly

Error Code	Description	Explanation	Remedy
		the drive moved the motor in order to detect it.	mounted, or the encoder ring is damaged. Please check the mounting instruction of your encoder for proper installation, and consult the encoder manufacturer for advices. 2. Inconsistency in signal transfer. Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influenced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging.

Error Code	Description	Explanation	Remedy
			Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
QeiOpnWr	QEI Open Wire.	Indicate that one of the differential A or B lines is disconnected. Only available on hardware that supports it.	Please check the wiring with for example the diode mode of a multimeter, making sure that the wires are stably transfering the signals.
Hall Sensor Faults			
HallSeq	Hall Sequence error.	The hall sensor outputs have an invalid value (111 or 000), or the sequence between the cur-	Please check the wiring and make sure that the signals are transfered with good con-

Error Code	Description	Explanation	Remedy
		rent state and the last state is invalid.	sistency. If the HALL sensor is used together with an incremental encoder for commutation, it can be that the HALL sensor A/B/C pins are connected in a wrong order. Please verify the cabling, and verify the order by running offset detection with only the HALL sensor configured for commutation. If the order is correct, the result of the offset detection should be around 0, and 0x2003:5 Motor phases inverted should show 0 (No).
A-Format sensor faults			

Error Code	Description	Explanation	Remedy
AftFrame	A-Format Frame error.	The fixed data in the recieved A-Format frame (like sync code) is wrong. This usually means that the A-Format transmission speed is wrong.	Please check the transmission speed configuration according to the encoder datasheet.
AftTmout	A-Format Timeout.	The A-Format encoder didn't start communication after sending a request.	Please check the wiring and make sure that the signals are transfered with good consistency. Use a oscilloscope to monitor the signal transfered for further investigation.
AftCrc	A-Format CRC error.	The A-Format frame contains a CRC of some of the data that is being sent. This error is raised when the CRC sent by the encoder is dif-	Please check the signals received by the drive with an oscilloscope under operation for better investigation. The cause can be that the signal transfer is influ-

Error Code	Description	Explanation	Remedy
		ferent than the CRC computed by the drive. This usually means that the drive can't read the frame properly, which usually happens because of noise in the communication link.	enced by external field, or that the signal transfer is unstable for example due to soldering quality or cable aging. Improve the situation by replacing the cables, better arranging the cables, adding shielding, etc., ensuring that the signals can be stably transfered with acceptable noise.
AftBatt	A-Format BATT status.	The encoder is raising Battery Alarm. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Battery Alarm. The usual case is that the battery voltage is low, the battery is wrongly wired, or there's no bat-

Error Code	Description	Explanation	Remedy
			tery connected.
AftMTErr	A-Format MTERR status.	The encoder is raising MT Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the MT Error.
AftOvFlw	A-Format OvFlow status.	The encoder is raising Over Flow Alarm. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Over Flow Alarm.
AftOvSpd	A-Format OVSPD status.	The encoder is raising Over Speed Alarm. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Over Speed Alarm. The usual case is that the rotation speed exceeds the speed limit of the encoder.
AftMemEr	A-Format MEMERR status.	The encoder is raising Memory	Please check the encoder doc-

Error Code	Description	Explanation	Remedy
		Error. Check your A-Format encoder datasheet for more information.	umentation for better understanding the meaning of the Memory Error.
AftSTErr	A-Format STERR status.	The encoder is raising ST Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the ST Error.
AftPSErr	A-Format PSERR status.	The encoder is raising PS Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the PS Error.
AftBusyF	A-Format BUSY status.	The encoder is raising BUSY Flag. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the BUSY Flag.
AftMemBs	A-Format MemBusy status.	The encoder is raising MEMBUSY Flag. Check your A-Format encoder	Please check the encoder documentation for better understanding the meaning of the

Error Code	Description	Explanation	Remedy
		datasheet for more information.	MEMBUSY Flag.
AftOvTmp	A-Format OvTemp status.	The encoder is raising Temperature Warning. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Temperature Warning. In case of over-temperature please refer to the remedies for over-temperature.
AftIncEr	A-Format IncErr status.	The encoder is raising Incremental Signal Error. Check your A-Format encoder datasheet for more information.	Please check the encoder documentation for better understanding the meaning of the Incremental Signal Error.
Generic Sensor Faults			
SnsrAngl	Sensor Angle error.	1. The electrical angle was not completeting a whole revolution	Please firstly check whether the motor rotates dur-

Error Code	Description	Explanation	Remedy
		during offset detection. 2. The calculated commutation angle is out of bounds. A singleturn related configuration parameter (resolution, singleturn) with a wrong value normally causes this error. 3. The motor is not moving during the QEI index detection.	ing offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. Please check the encoder configuration, and make sure that while rotating the motor for one full turn, the encoder value is correctly responding to the mechanical position. You can manually rotate the motor for investigation. An alternative way is with OBLAC opening another page and ploting the position feedback of the encoder used for commutation while running offset detection.

Error Code	Description	Explanation	Remedy
SnsrCfg	Sensor Configuration error.	1. Some encoder parameter was set with an invalid value in the Object dictionary. 2. An action that required a particular encoder function (position, velocity or commutation) was triggered, but there is no encoder configured for that function. 3. There was an issue when trying to initialize or configure an encoder.	Please check the configuration of the sensors, especially the parameters changed lately. Make sure that all functions required (position, velocity, commutation) are configured with the sensors.
HwRsrcEr	Hardware Resource Error.	The software configuration is invalid for the hardware. For example, a sensor configured is with a sensor type not supported by that port.	Please check the sensor configurations and the digital IO settings, making sure that the setting is supported by your hardware according to the doc-

Error Code	Description	Explanation	Remedy
			umentation.
IvldGpio	Invalid Gpio configuration.	The GPIO related configuration chosen is not available. This includes voltage level, direction of the GPIO, or GPIO trigger.	Please check the configuration of the GPIO according to the documentation.
SnsrBatt	Sensor battery error	This error is raised when an encoder battery was expected, but couldn't be detected. This error is raised only by multiturn encoders that need a battery, like Circulo's Internal encoder 1 when it's configured as a multiturn encoder.	Please check the documentation to make sure the battery is connected properly and power cycle. In case there is no battery or if the battery is not needed, select "0" or "No" to the "Multiturn resolution" parameter of this encoder.
SkpCycls	Serial encoder service is skipping cycles	This error is raised when a serial encoder service is looping	To try to solve either the warning or the error, we should try to spend

Error Code	Description	Explanation	Remedy
		slower than 16 kHz, which means that the cycles are taking longer than 62.5 us. This error should not be confused with the warning of the same name, which just means that an encoder that supports 32 kHz sampling (BiSS or SSI) is looping at 16 kHz instead.	less time reading the encoder. To do so we can first try to increase the clock frequency, and then try reducing the encoder timeout (check the encoder datasheet to determine which is the minimum value).
SwLimOut	Software Limit Out.	Target position is outside of the 0x607D: Software position limit defined by the user.	Please check object 0x607D: Software position limit and make sure that they are configured as expected. Accordingly send your position target.
Brake Faults			

Error Code	Description	Explanation	Remedy
BrkNtRls	Brake release failed	Brake release procedure for pin brake failed, i.e. the configured displacement threshold (0x2004:8 Minimum displacement (pin brake)) is not reached with the configrued torque (0x2004:9 Maximum torque (pin brake)).	Please firstly check whether the motor functions correct, for example by manually releasing the brake and running offset detection. If it doesn't please check its configuration, wiring, and mechanical mounting. While the motor functions correct, please further tune the two configuration parameters, by increasing the torque or decreasing the minimum displacement, until the brake can be released stably.
BkHiPull	Pull brake voltage too high	Pull brake voltage (0x2004:1 Pull voltage) higher than measured	Please measure the main power supply voltage sup-

Error Code	Description	Explanation	Remedy
		DC bus voltage (0x6079: DC link circuit voltage).	plied to the drive, and check whether the result is close to 0x6079: DC link circuit voltage. Pay attention that the actual voltage supplied to the drive would be slightly lower than the voltage output by the power supply, because of attenuation. Check the configuration of the pull/hold voltage, which should be lower than the DC Bus voltage.
BkHiHold	Hold brake voltage too high	Hold brake voltage (0x2004:2 Hold voltage) higher than measured DC bus voltage (0x6079: DC link circuit voltage).	
Open Terminal Faults			
OpnTermA	Open Terminal A	Drive terminal A is opened. This error can happen if motor cable is not connected to terminal A of the	Please check the motor wiring, making sure that the motor phase cable is connected to the terminal A/B/C of

Error Code	Description	Explanation	Remedy
		Drive. It can also happen if the Drive is damaged internally.	the Drive.
OpnTermB	Open Terminal B	Drive terminal B is opened. This error can happen if motor cable is not connected to terminal B of the Drive. It can also happen if the Drive is damaged internally.	
OpnTermC	Open Terminal C	Drive terminal C is opened. This error can happen if motor cable is not connected to terminal C of the Drive. It can also happen if the Drive is damaged internally..	
Open FET Faults			

Error Code	Description	Explanation	Remedy
OpnFetAH	Open high FET at phase A	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg A of the inverter, and it is the high-side FET of this leg.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
OpnFetBH	Open high FET at phase B	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg B of the inverter, and it is the high-side FET of this leg.	
OpnFetCH	Open high FET at phase C	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg C of the inverter, and it	

Error Code	Description	Explanation	Remedy
		is the high-side FET of this leg.	
OpnFetAL	Open low FET at phase A	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg A of the inverter, and it is the low-side FET of this leg.	
OpnFetBL	Open low FET at phase B	One of the power electronic switches of Drive is damaged (not conducting). The damaged FET is located in leg B of the inverter, and it is the low-side FET of this leg.	
OpnFetCL	Open low FET at phase C	One of the power electronic switches of Drive is damaged (not conducting). The	

Error Code	Description	Explanation	Remedy
		damaged FET is located in leg C of the inverter, and it is the low-side FET of this leg.	
Parameter Faults			
IvldOfst	Invalid offset	The commutation offset in 0x2001: Commutation angle offset is not valid. This error comes directly from reading object 0x2009:1 Commutation offset: State.	Please run offset detection before operations. Please pay attention that, the change in some fundamental configuration parameters (sensor configuration, motor configuration) can make the existing offset detection result invalid. Please redo offset detection in that case.
IvldOpmd	Invalid operational mode	Requested opmode is not supported. Set a valid value in 0x6060:	Please check our documentation for the available operation modes.

Error Code	Description	Explanation	Remedy
		Modes of operation.	Please check 0x6060: Modes of operation, and make sure that the master controller is sending a correct command.
ZeroMxI	Max. current is zero or negative	User-defined limit on motor max current (0x6073: Max current) is 0.	Please configure the 0x6073: Max current to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxT	Max. torque is zero or negative	User-defined limit on motor max current (0x6072: Max torque) is 0.	Please configure the 0x6072: Max torque to be a value larger than 0, according to your motor datasheet or your application need.
ZeroMxMS	Max. motor speed is zero or negative	The value of 0x6080: Max motor speed is 0, so no motion is allowed.	Pleaes configure 0x6080: Max motor speed to a valid value larger than 0, according

Error Code	Description	Explanation	Remedy
			to your motor data-sheet or your application need.
IvldPara	Invalid Parameter	The value of a parameter is invalid.	Please check the values changed lately according to the doc-umentation.
SiVelPfx	Invalid velocity scaling factor	The velocity scal-ing prefix defined by 0x60A9: SI unit velocity is outside of the supported range.	Please check OBLAC Drives, which directly provides available options for the con-figuration of 0x60A9: SI unit velocity, and the documentation for user defined velo-city units. Make sure that a valid data is write to the object.
MxPwrLmt	Maximum mech-anical output power of the motor is reached.	This warning appears when the limit has been reached that has been configured in	When this waring appears often in your setup and your application does not perform

Error Code	Description	Explanation	Remedy
		object 0x200B: Max power. The warning will disappear when the absolute value of reference torque has become less than this limit.	as intended please consider the following remedies: - set the limit higher if your application allows it - reconsider the size of your power-supply - choose a more powerful motor
Communication Faults			
CyclicHb	Cyclic heartbeat	A communication keep-alive heartbeat was dropped. Check that the PDO messages are being sent by the master cyclically, and within 100 ms of each other.	If the inactivity of communication is known/ intended, the occurrence of the error is expected. Do [Fault reset] to clear the fault after the communication is reestablished. Otherwise, the occurrence of the error indicates that

Error Code	Description	Explanation	Remedy
			there exists unknown issues in the communication chain. Possible causes include: - communication delay in master program, check the running status of the master machine. - disturbance in communication signal transfer, make sure that a communication cable with shielding is used or try with the cable provided by Synapticon, check the stability of cabling.
OsCmdCol	OS command collision	A new OS command was sent, but the OS command response of a previous OS command was not read after it com-	In order to send another OS command, please read the response of the previous OS command in the object 0x1023:3.

Error Code	Description	Explanation	Remedy
		pleted.	Whenever the first byte becomes 0, 1, 2 or 3, a new OS command can be sent.
I2t Protection Errors			
I2tActv	I2t protection is active	I2t protection is active, which limits your maximum torque to the nominal rated values of the motor. See 0x200A: I2t for configuration parameters.	Please check 0x200A: I2t and see whether it's configured as expected. Regarding the I2t being triggered, that indicates that the drive is protecting the motor from potential overheating. Please read our documentation of I2t for more information.
Homing errors			

Error Code	Description	Explanation	Remedy
HmInvMth	Unsupported or invalid homing method	A Homing method was selected that is not supported by SOMANET.	Please firstly check object 0x6098: Homing method and see whether the object is written as expected. Please refer to our documentation Homing modes, and check what homing methods are supported.
Internal Errors			
Intern01	Internal fault 1	Internal Error Code. Please contact support.	Please note down the information regarding the occurence of the error and reach Synapticon technical support for further investigation.
Intern02	Internal fault 2	Internal Error Code. Please contact support.	
Intern03	Internal fault 3	Internal Error Code. Please contact support.	

Error Code	Description	Explanation	Remedy
Filter cutoff frequency bounds			
VeFiFcLo	Velocity filter cut-off frequency too low	0x2021:2 Velocity feedback filter Cutoff frequency is lower than the minimum limit.	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
VeFiFcHi	Velocity filter cut-off frequency too high	0x2021:2 Velocity feedback filter Cutoff frequency is higher than the maximum limit.	Please check 0x2021:2 Velocity feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.

Error Code	Description	Explanation	Remedy
PoFiFcLo	Position filter cut-off frequency too low	0x2022:2 Position feedback filter Cutoff frequency is lower than the minimum limit.	Please check 0x2022:2 Position feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
PoFiFcHi	Position filter cut-off frequency too high	0x2022:2 Position feedback filter Cutoff frequency is higher than the maximum limit.	Please check 0x2022:2 Position feedback filter: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
VffFcLo	Velocity feed forward filter cut-off	0x2015:2 Velocity feed forward	Please check 0x2015:2 Velocity

Error Code	Description	Explanation	Remedy
	frequency too low	Cutoff frequency is lower than the minimum limit.	feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured higher than the limit 5 Hz. Please check the documentation of the object for latest information.
VffFcHi	Velocity feed forward filter cut-off frequency too high	0x2015:2 Velocity feed forward Cutoff frequency is higher than the maximum limit.	Please check 0x2015:2 Velocity feed forward: Cutoff frequency, and make sure that the cutoff frequency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
NhFiFcHi	Notch filter center frequency too high	0x2023:2 Notch filter Center frequency is higher than the maximum	Please check 0x2023 Notch filter, and make sure that the center fre-

Error Code	Description	Explanation	Remedy
		limit.	quency is configured lower than the limit 2000 Hz. Please check the documentation of the object for latest information.
NhFiPara	Invalid Notch filter parameters	Invalid configuration was detected for the notch filter.	Please check 0x2023 Notch filter, and make sure that the rejection band is set smaller than the center frequency.
Distributed Clock			
SynDifHi	Sync difference too high	Sync time variance of the EtherCAT distributed clock is too high (value will be ignored).This may happen if the distributed clock is poorly configured or the signal isn't consistent.	Please check the distrubuted clock (DC) setting of your EtherCAT master.

Error Code	Description	Explanation	Remedy
Temperature Sensor Faults			
TmpCore		This fault is triggered when the temperature of the core becomes too high and crosses the error threshold (> 100 °C). This happens usually because of sustained load on the processor, high operating frequency and even high ambient temperature. High core temperature can lead to equipment damage. It can also result in irregular clock timing.	Throttling the core is usally the remedy but as this is not possible in the case of real-time critical applications like ours, the solution would be to disable the drive until the reason for the error is resolved or use external cooling.
TmpDrive		This fault is triggered when the temperature of the drive becomes	Disable the drive until the reason for the error is resolved or use external cooling to

Error Code	Description	Explanation	Remedy
		too high and crosses the error threshold (> 100 °C). This happens usually because of sustained load on the processor, current leakage from the on-board componenets, high operating frequency and even ambient temperature. High drive temperature can lead to equipment damage.	bring down the over-all temperature.

0x2040 Input counter

Unsigned 32bit integer, resets after 4,294,967,295 (2^32 - 1) µs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Input counter	0x2040:0	UDINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x20E1 High resolution data

Prototype - This object will change in future releases!

High-resolution data contains all position, velocity and phase current measurements of the last millisecond. It is updated once per millisecond with the new data.

There are four sub-objects, which, when concatenated, reflect the full chunk of data. The timestamp corresponds to the measurements marked "1". Time step between position/velocity values is the motion control loop period, time step between phase currents is the torque control loop period. The data is encoded as follows:

Bytes	Value
0-3	Timestamp (32-bit unsigned int)
4-7	Position 1 (32-bit signed int)
8-11	Position 2 (32-bit signed int)
12-15	Position 3 (32-bit signed int)
16-19	Position 4 (32-bit signed int)
20-23	Velocity 1 (32-bit signed int)
24-27	Velocity 2 (32-bit signed int)
28-31	Velocity 3 (32-bit signed int)
32-35	Velocity 4 (32-bit signed int)
36-37	Phase Current I_b 1 (16-bit signed int)
38-39	Phase Current I_c 1 (16-bit signed int)
40-41	Phase Current I_b 2 (16-bit signed int)

Bytes	Value
42-43	Phase Current I_c 2 (16-bit signed int)
...	...
96-97	Phase Current I_b 16 (16-bit signed int)
98-99	Phase Current I_c 16 (16-bit signed int)

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 1	0x20E1:1	OCTET_STRING (25)	200			0		read-only (default)	Transmit PDO (Inputs)
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 2	0x20E1:2	OCTET_STRING	200			0		read-only (default)	Transmit PDO (Inpu-

		(25)							ts)
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 3	0x20-E1:3	OCTET_STRING (25)	20-0			0		read-only (default)	Transmit PDO (Inputs)
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Data 4	0x20-E1:4	OCTET_STRING (25)	20-0			0		read-only (default)	Transmit PDO (Inputs)

0x20F0 Timestamp

Position signal measurement timestamp. Unsigned 32bit integer, resets after 42,949,672 ((2^32 - 1) / 100) µs.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Time-	0x20-	UD-	32	0	42949-	0	µs	read-	Tran-

stamp	F0:0	INT			67295			only (default)	smit PDO (Inputs)

0x20F2 Assigned name

Allows a unique name to be assigned to this drive. This is only for reference; it is not used by the firmware.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Assigned name	0x20F2:0	STRING (50)	400					read-write	

0x20F3 DC synchronization

Indicates the status of the selected network synchronization mechanism.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
DC pulse time	0x20F3:1	UDINT	32			0		read-write	
The time when the last DC pulse happened in clock ticks (10 ns).									

0x2110 Encoder 1 configuration

All configuration options for Encoder 1

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Sensor port	0x2110:1	USINT	8	No sensor port: 0 Sensor port 1: 1 Sensor port 2: 2 Sensor port 3: 3 Sensor port 4: 4 Sensor port 5: 5		readwrite	

All encoders

The sensor port on which this encoder is connected.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping

			e				ng
Type	0x21-10:2	US-IN-T	8	None 0 Hall sensor 1 Incremental encoder 2 Nikon A-format 3 BiSS 4 REM 16MT 6 SSI 7 SinCos module 9		rea-dwri-te	

All encoders

Select the type of encoder you want to use.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Resolu-	0x21-10:3	U-DI-	3-2	1		1	Inc/Revolu-	rea-dwri-	

tion		NT					tion	te	

BiSS/SSI

Number of pulses in one revolution.
Most of BiSS/SSI encoders have a binary resolution, which means:
Resolution = 2^(subindex 12 *Singleturn bits*).

Incremental encoder

Total number of quadrature edges per revolution.
Resolution = Cycles per Revolution * 4

Hall sensor/Nikon A-Format/REM16MT/SinCos module

For these encoders this parameter is configured automatically.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Zero velocity threshold	0x2110:4	UDINT	32	0	1000000	4000	µs	readwrite	

Incremental encoder

This parameter is only used for the velocity calculation of incremental encoders.
If no new tick is received from the encoder during the time specified in this parameter, the velocity of the encoder will be set to 0.
The value will automatically be rounded down to a multiple of 62.5 us.

Name	Index:Sub	Type	Bit S-	Options	Unit	Access	PDO Ma-

			iz-e				ppi-ng
Polar-ity	0x21-10:5	B-O-OL	1	Normal polarity: 0 Inverted polarity: 1		rea-dwri-te	

All encoders

Invert the direction of movement of the encoder. This is the equivalent of physically flipping the encoder, so the commutation angle offset will not be valid after changing this parameter. If you have an axis with two encoders (one for commutation and one for position and velocity), the direction of each encoder must match for control stability. This polarity parameter can be used for flipping the direction of the position or velocity encoder relative to the sensor used for commutation.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
Sing-leturn offset	0x21-10:6	U-DI-NT	3-2			0	Inc	rea-dwri-te	

Absolute encoders (BiSS/SSI/Nikon A-format/REM 16MT)

This value is substracted from the singleturn value received from the encoder. It is the equivalent of physically moving the encoder, so the commutation angle offset will not be valid after changing this parameter. This parameter has to be lower than the resolution of the encoder.
The singleturn offset is applied after the subindex 5 *Polarity* has been applied.

Nam-	Inde-	Ty-	B-	Options	Unit	Acc-	PD-

e	x:Sub	pe	it Size			ess	O Mapping
Access signal type	0x21-10:7	BOOL	1	Differential RS422	0	readwrite	
				Single-ended TTL	1		

Incremental encoder

Select if the encoder signal is differential (RS422) or single-ended (TTL).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Clock frequency	0x21-10:8	UINT	1-6	1000	16000	1000	kHz	readwrite	

BiSS/SSI

Frequency of the SSI clock line or BiSS MA line.
Refer to your encoder datasheet to check what are the minimum and maximum values for this parameter.
The drive can only clock at some discrete frequencies. If the user enters a frequency that is not within these discrete frequencies, it will be set to the next possible lower frequency.
To calculate what's the real frequency that will be used, you can round up the result of

25000 / Clock frequency. Then divide 25000 by the rounded value obtained before.
For example, when setting 4000 KHz the real frequency will be 3571.43 kHz.
The maximum value that can be set is 5000 kHz.

Nikon A-format

Select the transmission speed of the encoder. The only possible values are 2500, 4000, 6670, 8000 and 16000 kbps.
See encoder datasheet for more information.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Frame size	0x2110:9	USINT	8	1		1		readwrite	

BiSS

The BiSS service expects an encoder that has the following frame structure:
- SLO line high when encoder is ready to start communicating.
- Ack bit (which can be extended to multiple bits by the encoder).
- Start bit when the encoder is ready to transmit the data.
- CDS bit for register communication.
- Payload (any data can be sent here, but the only data that can be processed is Multiturn and singleturn position).
- Error and warning bits, in this order.
- Variable length CRC.
- Timeout.

This parameter specifies the number of bits transmited as Payload.

SSI

This parameter specifies the number of bits transmited in one encoder frame.

Name	Index	Type	Bit	Min	Max	Default	Unit	Access	PDO

e	x:Sub	pe	it Size	Data	Data	ault Data		ess	O Mapping
Multi-turn bits	0x21-10:1-0	USINT	8			0		read-write	

BiSS/SSI/Nikon A-format

Number of multiturn bits to read from the encoder.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Multi-turn first bit position	0x21-10:1-1	USINT	8			0		read-write	

BiSS/SSI

Position of the first bit of multiturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*.
Usually this value is 0, since multiturn is the first element on the frame.
Only relevant if subindex 10 *Multiturn bits* is bigger than 0.

Name	Index:Sub	Type	Bit S-	Min Data	Max Data	Default Dat-	Unit	Access	PDO Ma-

			iz-e			a			ppi-ng
Sing-leturn bits	0x21-10:1-2	US-IN-T	8	1		1		rea-dwri-te	

BiSS/SSI

Number of singleturn bits to read from the encoder.

Nikon A-format

Number of singleturn bits to read from the encoder. The only possible values are 17, 20, 22 and 24.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
Sing-leturn first bit pos-ition	0x21-10:1-3	US-IN-T	8			0		rea-dwri-te	

BiSS/SSI

Position of the first bit of singleturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*. For most of the BiSS/SSI encoders, this value is equal to: subindex 10 *Multiturn bits* + subindex 11 *Multiturn first bit position*.

Nam-e	Inde-x:Su-	Ty-pe	B-it	Min Dat-	Max Dat-	Def-ault	Unit	Acc-ess	PD-O

	b		S-iz-e	a	a	Dat-a			Ma-ppi-ng
Time-out	0x21-10:1-4	US-IN-T	8			36	µs	rea-dwri-te	

BiSS/SSI

Timeout that will be sent by the drive at the end of each BiSS/SSI frame, which informs the encoder that the frame is over.
Refer to your encoder datasheet to check what is the minimum value for this parameter.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
CRC poly-nomi-al	0x21-10:1-5	US-IN-T	8			67		rea-dwri-te	

BiSS

Polynomial for the CRC calculation of the sensor data (SCD).
This value is obtained by writing the polynomial in binary representation.
For example: x^6 + x^3 + x^1 + x^0 is 0b1001011 = 0x4B = 75.
The default value of 67 (x^6 + x^1 + x^0 = 0b1000011 = 0x43) is the one used for most of the BiSS encoders.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-

			e						ng
Maximum tbusy	0x2110:16	USINT	8			20	µs	readwrite	

BiSS

This value defines the maximum time to wait for the Start bit once the Ack bit was already received. If the Start bit is not sent after this time, the error BisSrtBt is raised.
Check your encoder datasheet to find what's the typical or maximum value for this parameter.
It is advised to try to increase this number in case the error BisSrtBt appears, which is likely to happen because the encoder needed more time to process the data before sending the Start bit.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Status bits active value	0x2110:17	USINT	8	Status bits active low: 0 Status bits active high: 1		readwrite	

BiSS

Value of the status bits when active. The BiSS-C standard recommends that these bits are active low, so most encoders should work with the default value.

Name	Index:Sub	Type	B-	Options	Unit	Access	PD-

	b		it S-iz-e				O Ma-ppi-ng
Par-ity type	0x21-10:1-8	US-IN-T	8	No par-ity: 0 Odd parity: 1 Even parity: 2		rea-dwri-te	

SSI

Type of parity read from the encoder. If there is parity, it should be the last bit on the SSI frame.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
First clock delay	0x21-10:1-9	UI-NT	1-6			0	µs	rea-dwri-te	

SSI

Delay after first clock edge. This is compatible only with some encoders (will be specified in the encoder datasheet). If your encoder supports this function, it's normally used to be able to increase the maximum allowable clock frequency.

Nam-	Inde-	Ty-	B-	Options	Unit	Acc-	PD-

e	x:Sub	pe	it Size			ess	O Mapping
Data ordering	0x2110:20	BOOL	1	Natural Binary Code: 0 Reflected Binary Code: 1		readwrite	

SSI

Select if the ordering of position is done using Natural Binary Code or Reflected Binary Code (also known as Gray code).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Endianness	0x2110:21	BOOL	1	MSB first: 0 LSB first: 1		readwrite	

SSI

Order of the bits (Most Significant Bit or Least Significant Bit transmitted first). Usually MSB is transmitted first.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Index availability	0x2110:22	BOOL	1	Encoder without index: 0 Encoder with index: 1		readwrite	

Incremental encoder/SinCos module

Define if the connected encoder has an index or not.
For SinCos encoders the index is often called "reference".
Incremental encoders without index are often called AB encoders, while incremental encoders with index are often called ABI encoders.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Hall sensor port	0x2110:23	USINT	8	No sensor port: 0 Sensor port 1: 1		readwrite	

				<table><tr><td>Sensor port 2</td><td>2</td></tr><tr><td>Sensor port 3</td><td>3</td></tr><tr><td>Sensor port 4</td><td>4</td></tr><tr><td>Sensor port 5</td><td>5</td></tr></table>					

Incremental encoder

This parameter is used to define to which Sensor port the Hall sensor is connected in case of using it together with an incremental encoder in order to have an approximate absolute position while the incremental encoder index is not yet found.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Sinewave cycles per revolution	0x2110:24	UINT	16	1		1		readwrite	

SinCos module

Number of sinewaves cycles in one motor revolution.
This value is going to be used to calculate the singleturn resolution (subindex 3 *Resolution*).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Sinewave resolution	0x2110:25	USINT	8	8 13 16 12 32 11 64 10 128 9 256 8 512 7 1024 6 2048 5 4096 4 8192 3	Inc	readwrite	

SinCos module

Select desired number of pulses in one sinewave.
This value is going to be used to calculate the singleturn resolution (sunindex 3 *Resolution*).

Name	Index:Sub	Type	Bit Siz-	Options	Unit	Access	PDO Mappi-

			e				ng
Sinusoidal output voltage	0x21-10:2-6	USINT	8	up to 50 mVpp: 15 up to 60 mVpp: 14 up to 75 mVpp: 13 up to 0.1 Vpp: 12 up to 0.12 Vpp: 11 up to 0.14 Vpp: 10 up to 0.15 Vpp: 9 up to 0.2 Vpp: 8		read-write	

up to 0.28 Vpp	7
up to 0.4 Vpp	6
up to 0.5 Vpp	5
up to 0.6 Vpp	4
up to 0.75 Vpp	3
up to 1 Vpp	2
up to 1.2 Vpp	1
up to 1.5 Vpp	0

SinCos module

Differential voltage in the Sin/Cos outputs peak to peak.
It is often abreviated as Vpp.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Filter	0x2110:27	USINT	8	Disabled: 0 Filter strength 2 (weakest): 2 Filter strength 3: 3 Filter strength 4: 4 Filter strength 5: 5 Filter strength 6: 6 Filter strength 7: 7 Filter: 8		readwrite	

				strength 8				
				Filter strength 9 (stronge-st)	9			

REM 16MT

The filter with strength F of the sensor. The estimated step response time of the filter is Ts*2^F (Ts is the sampling time, ca. 70µs).

0x2111 Encoder 1 feedback

Feedback of the encoder configured as Encoder 1

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Raw position	0x2111:1	UDINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)
The position value directly read from the encoder.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Adjusted position	0x211-1:2	DINT	32			0		read-only (default)	Transmit PDO (Inputs)
The position value after applying the Singleturn offset and the Sensor polarity and the location of the index. This is an absolute value.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity	0x211-1:3	DINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x2112 Encoder 2 configuration

All configuration options for Encoder 2

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Sensor port	0x2112:1	USINT	8	No sensor port	0		readwrite	

				Sensor port 1: 1 Sensor port 2: 2 Sensor port 3: 3 Sensor port 4: 4 Sensor port 5: 5			

All encoders

The sensor port on which this encoder is connected.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Type	0x2112:2	USINT	8	None: 0 Hall sensor: 1 Incremental encoder: 2 Nikon A-: 3		readwrite	

				<table><tr><td>format</td><td></td></tr><tr><td>BiSS</td><td>4</td></tr><tr><td>REM 16MT</td><td>6</td></tr><tr><td>SSI</td><td>7</td></tr><tr><td>SinCos module</td><td>9</td></tr></table>				

All encoders

Select the type of encoder you want to use.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Resolution	0x21-12:3	UDINT	32	1		1	Inc/Revolution	readwrite	

BiSS/SSI

Number of pulses in one revolution.
Most of BiSS/SSI encoders have a binary resolution, which means:
Resolution = 2^(subindex 12 *Singleturn bits*).

Incremental encoder

Total number of quadrature edges per revolution.
Resolution = Cycles per Revolution * 4

Hall sensor/Nikon A-Format/REM16MT/SinCos module

For these encoders this parameter is configured automatically.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Zero velocity threshold	0x2112:4	UDINT	32	0	1000000	4000	µs	readwrite	

Incremental encoder

This parameter is only used for the velocity calculation of incremental encoders.
If no new tick is received from the encoder during the time specified in this parameter, the velocity of the encoder will be set to 0.
The value will automatically be rounded down to a multiple of 62.5 us.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Polarity	0x2112:5	BOOL	1	Normal polarity: 0 Inverted polarity: 1		readwrite	

All encoders

Invert the direction of movement of the encoder. This is the equivalent of physically flipping

the encoder, so the commutation angle offset will not be valid after changing this parameter. If you have an axis with two encoders (one for commutation and one for position and velocity), the direction of each encoder must match for control stability. This polarity parameter can be used for flipping the direction of the position or velocity encoder relative to the sensor used for commutation.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Singleturn offset	0x2112:6	UDINT	32			0	Inc	readwrite	

Absolute encoders (BiSS/SSI/Nikon A-format/REM 16MT)

This value is substracted from the singleturn value received from the encoder. It is the equivalent of physically moving the encoder, so the commutation angle offset will not be valid after changing this parameter. This parameter has to be lower than the resolution of the encoder.
The singleturn offset is applied after the subindex 5 *Polarity* has been applied.

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Access signal type	0x2112:7	BOOL	1	Differential RS422	0		readwrite	
				Single-	1			

				ended TTL					

Incremental encoder

Select if the encoder signal is differential (RS422) or single-ended (TTL).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Clock frequency	0x2112:8	UINT	16	1000	16000	1000	kHz	readwrite	

BiSS/SSI

Frequency of the SSI clock line or BiSS MA line.
Refer to your encoder datasheet to check what are the minimum and maximum values for this parameter.
The drive can only clock at some discrete frequencies. If the user enters a frequency that is not within these discrete frequencies, it will be set to the next possible lower frequency.
To calculate what's the real frequency that will be used, you can round up the result of 25000 / Clock frequency. Then divide 25000 by the rounded value obtained before.
For example, when setting 4000 KHz the real frequency will be 3571.43 kHz.
The maximum value that can be set is 5000 kHz.

Nikon A-format

Select the transmission speed of the encoder. The only possible values are 2500, 4000, 6670, 8000 and 16000 kbps.
See encoder datasheet for more information.

Name	Index:Sub	Type	B-	Min Dat-	Max Dat-	Def-	Unit	Access	PD-

	b		it S-iz-e	a	a	ault Dat-a			O Ma-ppi-ng
Fram-e size	0x21-12:9	US-IN-T	8	1		1		rea-dwri-te	

BiSS

The BiSS service expects an encoder that has the following frame structure:
- SLO line high when encoder is ready to start communicating.
- Ack bit (which can be extended to multiple bits by the encoder).
- Start bit when the encoder is ready to transmit the data.
- CDS bit for register communication.
- Payload (any data can be sent here, but the only data that can be processed is Multiturn and singleturn position).
- Error and warning bits, in this order.
- Variable length CRC.
- Timeout.

This parameter specifies the number of bits transmited as Payload.

SSI

This parameter specifies the number of bits transmited in one encoder frame.

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Min Dat-a	Max Dat-a	Def-ault Dat-a	Unit	Acc-ess	PD-O Ma-ppi-ng
Multi-turn bits	0x21-12:1-0	US-IN-T	8			0		rea-dwri-te	

BiSS/SSI/Nikon A-format

Number of multiturn bits to read from the encoder.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Multiturn first bit position	0x2112:1	USINT	8			0		readwrite	

BiSS/SSI

Position of the first bit of multiturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*.
Usually this value is 0, since multiturn is the first element on the frame.
Only relevant if subindex 10 *Multiturn bits* is bigger than 0.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Singleturn bits	0x2112:2	USINT	8	1		1		readwrite	

BiSS/SSI

Number of singleturn bits to read from the encoder.

Nikon A-format

Number of singleturn bits to read from the encoder. The only possible values are 17, 20, 22 and 24.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Singleturn first bit position	0x2112:13	USINT	8			0		readwrite	

BiSS/SSI

Position of the first bit of singleturn position on the frame. Position 0 of the frame corresponds to the first bit transmitted within the frame size specified in subindex 9 *Frame size*. For most of the BiSS/SSI encoders, this value is equal to: subindex 10 *Multiturn bits* + subindex 11 *Multiturn first bit position*.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Timeout	0x2112:14	USINT	8			36	µs	readwrite	

BiSS/SSI

Timeout that will be sent by the drive at the end of each BiSS/SSI frame, which informs the encoder that the frame is over.

Refer to your encoder datasheet to check what is the minimum value for this parameter.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
CRC polynomial	0x2112:15	USINT	8			67		readwrite	

BiSS

Polynomial for the CRC calculation of the sensor data (SCD).
This value is obtained by writing the polynomial in binary representation.
For example: x^6 + x^3 + x^1 + x^0 is 0b1001011 = 0x4B = 75.
The default value of 67 (x^6 + x^1 + x^0 = 0b1000011 = 0x43) is the one used for most of the BiSS encoders.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Maximum tbusy	0x2112:16	USINT	8			20	µs	readwrite	

BiSS

This value defines the maximum time to wait for the Start bit once the Ack bit was already received. If the Start bit is not sent after this time, the error BisSrtBt is raised.
Check your encoder datasheet to find what's the typical or maximum value for this parameter.

It is advised to try to increase this number in case the error BisSrtBt appears, which is likely to happen because the encoder needed more time to process the data before sending the Start bit.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Status bits active value	0x2112:17	USINT	8	Status bits active low: 0 Status bits active high: 1		read-write	

BiSS

Value of the status bits when active. The BiSS-C standard recommends that these bits are active low, so most encoders should work with the default value.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Parity type	0x2112:18	USINT	8	No parity: 0 Odd parity: 1		read-write	

				Even parity: 2				

SSI

Type of parity read from the encoder. If there is parity, it should be the last bit on the SSI frame.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
First clock delay	0x2112:19	UINT	16			0	µs	readwrite	

SSI

Delay after first clock edge. This is compatible only with some encoders (will be specified in the encoder datasheet). If your encoder supports this function, it's normally used to be able to increase the maximum allowable clock frequency.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Data ordering	0x2112:20	BOOL	1	Natural Binary Code: 0 Reflec-: 1		readwrite	

				ted Binary Code			

SSI

Select if the ordering of position is done using Natural Binary Code or Reflected Binary Code (also known as Gray code).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Endianness	0x2112:21	BOOL	1	MSB first: 0 LSB first: 1		readwrite	

SSI

Order of the bits (Most Significant Bit or Least Significant Bit transmitted first). Usually MSB is transmitted first.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Index availabil-	0x2112:22	BOOL	1	Encoder without: 0		readwrite	

ity				index Encoder with index: 1			

Incremental encoder/SinCos module

Define if the connected encoder has an index or not.
For SinCos encoders the index is often called "reference".
Incremental encoders without index are often called AB encoders, while incremental encoders with index are often called ABI encoders.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Hall sensor port	0x2112:23	USINT	8	No sensor port: 0 Sensor port 1: 1 Sensor port 2: 2 Sensor port 3: 3 Sensor port 4: 4		readwrite	

				Sensor port 5 | 5				

Incremental encoder

This parameter is used to define to which Sensor port the Hall sensor is connected in case of using it together with an incremental encoder in order to have an approximate absolute position while the incremental encoder index is not yet found.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Sinewave cycles per revolution	0x2112:24	UINT	16	1		1		readwrite	

SinCos module

Number of sinewaves cycles in one motor revolution.
This value is going to be used to calculate the singleturn resolution (subindex 3 *Resolution*).

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Sinewave res-	0x2112:25	USINT	8	8 | 13	Inc	readwrite	

olu-tion				<table><tr><td>16</td><td>12</td></tr><tr><td>32</td><td>11</td></tr><tr><td>64</td><td>10</td></tr><tr><td>128</td><td>9</td></tr><tr><td>256</td><td>8</td></tr><tr><td>512</td><td>7</td></tr><tr><td>1024</td><td>6</td></tr><tr><td>2048</td><td>5</td></tr><tr><td>4096</td><td>4</td></tr><tr><td>8192</td><td>3</td></tr></table>			

SinCos module

Select desired number of pulses in one sinewave.
This value is going to be used to calculate the singleturn resolution (sunindex 3 *Resolution*).

Nam-e	Inde-x:Su-b	Ty-pe	B-it S-iz-e	Options	Unit	Acc-ess	PD-O Ma-ppi-ng
Sinus-oidal out-put volta-ge	0x21-12:2-6	US-IN-T	8	<table><tr><td>up to 50 mVpp</td><td>15</td></tr><tr><td>up to</td><td>14</td></tr></table>		rea-dwri-te	

60 mVpp	
up to 75 mVpp	13
up to 0.1 Vpp	12
up to 0.12 Vpp	11
up to 0.14 Vpp	10
up to 0.15 Vpp	9
up to 0.2 Vpp	8
up to 0.28 Vpp	7
up to 0.4 Vpp	6

				up to 0.5 Vpp: 5 up to 0.6 Vpp: 4 up to 0.75 Vpp: 3 up to 1 Vpp: 2 up to 1.2 Vpp: 1 up to 1.5 Vpp: 0			

SinCos module

Differential voltage in the Sin/Cos outputs peak to peak.
It is often abreviated as Vpp.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping

Filter	0x21-12:2-7	US-IN-T	8	Disabled: 0 Filter strength 2 (weakest): 2 Filter strength 3: 3 Filter strength 4: 4 Filter strength 5: 5 Filter strength 6: 6 Filter strength 7: 7 Filter strength 8: 8 Filter strength 9: 9		read-write	

				(stronge-st)				

REM 16MT

The filter with strength F of the sensor. The estimated step response time of the filter is Ts*2^F (Ts is the sampling time, ca. 70µs).

0x2113 Encoder 2 feedback

Feedback of the encoder configured as Encoder 2

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Raw position	0x2113:1	UDINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)
The position value directly read from the encoder.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Adjusted position	0x2113:2	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

The position value after applying the Singleturn offset and the Sensor polarity and the location of the index. This is an absolute value.									
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity	0x2113:3	DINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x2210 GPIO pin configuration

Configurable general purpose inputs and outputs with a 4 kHz update rate.

For hardware that supports it, each pin can be configured as follows:

Value	Definition
0	Disabled
2	[IN] General Purpose Input
3	[OUT] General Purpose Output
4	[IN] Home Switch Active High - Level Triggered
5	[IN] Home Switch Active Low - Level Triggered
6	[IN] Positive Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)
7	[IN] Positive Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (PosLmAct)

8	[IN] Negative Limit Switch Active High - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
9	[IN] Negative Limit Switch Active Low - Level Triggered for Homing & Edge Triggered (Latch) for Error (NegLmAct)
10	[IN] Interlock (Disable Operation) Active Low - Level Triggered
11	[IN] Fault Reset - Edge Triggered
12	[IN] Interlock (Fault Reaction) Active Low - Level Triggered
13	[IN] Time Stamped Rising Edge - Edge Triggered
14	[OUT] Mimics Target Reached (tr) Bit of Statusword
15	[OUT] Mimics Internal Limit Active (ila) Bit of Statusword
16	[OUT] Timestamped Triggered Output
17	[OUT] Position Triggered Output (Pulse)
18	[OUT] Mimics Fault (f) Bit of Statusword
19	[OUT] Mimics Switched On (so) Bit of Statusword
20	[OUT] Mimics Ready to Switch On (rtso) Bit of Statusword
21	[OUT] Mimics Operation Enabled (oe) Bit of Statusword
22	[OUT] Mimics Voltage Enabled (ve) Bit of Statusword
23	[OUT] Mimics Switch On Disabled (sod) Bit of Statusword
24	[OUT] Mimics Quick Stop (qs) Bit of Statusword
25	[OUT] Mimics Warning (w) Bit of Statusword

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 1	0x2210:1	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Position Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

				Voltage Enabled [OUT] Switch On Disabled: 23 [OUT] Quick Stop: 24 [OUT] Warning: 25 [OUT] NZR LED: 26			

Pin 1 : DIO [With Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 2	0x2210:2	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Act-		read-write	

				ive High)			
				[IN] Home Switch (Active Low)	5		
				[IN] Limit Switch Positive (Active High)	6		
				[IN] Limit Switch Positive (Active Low)	7		
				[IN] Limit Switch Negative (Active High)	8		
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault	11		

Reset (Active High)	
[IN] Interlock - Fault Reaction (Active Low)	12
[IN] Timestamped Rising Edge (Active High)	13
[OUT] Target Reached	14
[OUT] Internal limit Active	15
[OUT] Timestamped Triggered Output	16
[OUT] Position Triggered Output	17

(Pulse)	
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT] Voltage Enabled	22
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 2 : DIO [With Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 3	0x2210:3	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Position Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

				Voltage Enabled [OUT] Switch On Disabled 23 [OUT] Quick Stop 24 [OUT] Warning 25 [OUT] NZR LED 26			
Pin 3 : DIO [With Pull Down Resistor]							
Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 4	0x22-10:4	USINT	8	Disabled 0 [IN] General Purpose 2 [OUT] General Purpose 3 [IN] Home Switch (Act-		read-write	

				ive High)			
				[IN] Home Switch (Active Low)	5		
				[IN] Limit Switch Positive (Active High)	6		
				[IN] Limit Switch Positive (Active Low)	7		
				[IN] Limit Switch Negative (Active High)	8		
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault	11		

				Reset (Active High)			
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target Reached	14		
				[OUT] Internal limit Active	15		
				[OUT] Timestamped Triggered Output	16		
				[OUT] Position Triggered Output	17		

(Pulse)	
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT] Voltage Enabled	22
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 4 : DIO [With Pull Down Resistor] (For Circulo : Only DO)

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 5	0x2210:5	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Position Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

				Voltage Enabled [OUT] Switch On Disabled: 23 [OUT] Quick Stop: 24 [OUT] Warning: 25 [OUT] NZR LED: 26			

Pin 5 : DIO [Without Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 6	0x2210:6	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Act-		read-write	

				ive High)			
				[IN] Home Switch (Active Low)	5		
				[IN] Limit Switch Positive (Active High)	6		
				[IN] Limit Switch Positive (Active Low)	7		
				[IN] Limit Switch Negative (Active High)	8		
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault	11		

Reset (Active High)	
[IN] Interlock - Fault Reaction (Active Low)	12
[IN] Timestamped Rising Edge (Active High)	13
[OUT] Target Reached	14
[OUT] Internal limit Active	15
[OUT] Timestamped Triggered Output	16
[OUT] Position Triggered Output	17

(Pulse)	
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT] Voltage Enabled	22
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 6 : DIO [Without Pull Down Resistor]

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Pin 7	0x2210:7	USINT	8	Disabled: 0 [IN] General Purpose: 2 [OUT] General Purpose: 3 [IN] Home Switch (Active High): 4 [IN] Home Switch (Active Low): 5 [IN] Limit Switch Positive (Active High): 6 [IN] Limit Switch Positive (Active Low): 7 [IN] Limit Switch Neg-: 8		read-write	

				ative (Active High)			
				[IN] Limit Switch Negative (Active Low)	9		
				[IN] Interlock - Disable Operation (Active Low)	10		
				[IN] Fault Reset (Active High)	11		
				[IN] Interlock - Fault Reaction (Active Low)	12		
				[IN] Timestamped Rising Edge (Active High)	13		
				[OUT] Target	14		

Reached	
[OUT] Internal limit Active	15
[OUT] Timestamp-ed Triggered Output	16
[OUT] Position Triggered Output (Pulse)	17
[OUT] Fault	18
[OUT] Switched On	19
[OUT] Ready to Switch On	20
[OUT] Operation Enabled	21
[OUT]	22

Voltage Enabled	
[OUT] Switch On Disabled	23
[OUT] Quick Stop	24
[OUT] Warning	25
[OUT] NZR LED	26

Pin 7 : DI [With Pull Down Resistor]

0x2211 GPIO output events

Configuration to set the value in the *Digital Output* object to the corresponding GPIO when a certain event happens.

The GPIO has to be enabled in *0x2210 GPIO*.

Configurable events are:

`Timestamp triggered`: Output at a certain timestamp.

`Target reached`: Output when target is reached. This mimics bit 10 (tr) of the Statusword.

`Internal limit active`: Output when internal limit is active. This mimics bit 11 (ila) of the Statusword.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Timestamp set-point	0x22-11:1	UDINT	32			0		read-write	Receive PDO (Outputs)
Timestamp at which the GPIO should trigger in case "Timestamp triggered" is selected. The unit is in 10 ns.									

0x2212 GPIO input actions

Configure an input GPIO to perform a certain action depending on its value.

The GPIO has to be enabled in *0x2210 GPIO*.

Configurable actions are:

`Timestamped rising edge`: Report timestamp of selected GPIO last rising edge.

`Disable operation`: Clear bit 3 (eo) of controlword when the selected GPIO is high.

`Quick stop`: Clear bit 2 (qs) of controlword when the selected GPIO is high.

`Fault reset`: Set bit 7 (fr) of controlword when the selected GPIO is high.

Name	Index-:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Timestamp on rising edge	0x22-12:1	UDINT	32			0		read-only (default)	Transmit PDO (Inputs)
Timestamp at which the last rising edge happened in case "Timestamped rising edge" is selected. The unit is in 10 ns.									

0x2213 GPIO position trigger

Configure a GPIO to output a width-configurable pulse everytime certain encoder position is reached.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position setpoint	0x2213:1	DINT	32			0	Inc	read-write	Receive PDO (Outputs)
Position value at which the GPIO should be triggered.									

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Target encoder	0x2213:2	USINT	8	No encoder selected	0		read-write	
				Encoder 1	1			
				Encoder 2	2			
Encoder whose position is used for this feature.								

Name	Index:Sub	Type	Bit Size	Min Data	Max Dat-	Default Dat-	Unit	Access	PDO Mapping

					a	a			
Pulse width	0x22-13:3	USINT	8			0	ms	read-write	
Width of the pulse when the GPIO is triggered.									

0x2214 GPIO global options

undefined

Name	Index:-Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Voltage level	0x2214:1	USINT	8	3.3 volts	1		read-write	
				5.0 volts	2			
Select the global voltage level of GPIO ports.								

0x2401 Analog input 1

Analog Input value of SOMANET Drive if available.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 1	0x2401:0	UINT	16			0		read-only (default)	Transmit PDO (Inpu-

									ts)

0x2402 Analog input 2

Analog Input value of SOMANET Drive if available.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 2	0x2402:0	UINT	16			0		read-only (default)	Transmit PDO (Inputs)

0x2403 Analog input 3

Analog Input value of SOMANET Drive if available.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Analog input 3	0x2403:0	UINT	16			0		read-only (default)	Transmit PDO (Inputs)

0x2404 Analog input 4

Analog Input value of SOMANET Drive if available.

Na-	Index:-	Ty-	Bit	Mi-	Ma-	Def-	Un-	Acce-	PDO

me	Sub	pe	Si-ze	n Da-ta	x Da-ta	ault Data	it	ss	Map-ping
Ana-log inpu-t 4	0x240-4:0	UI-NT	16			0		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x2611 Safety digital input diagnostics

Digital Inputs on Safety Module

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	Un-it	Acce-ss	PDO Map-ping
Inp-ut 1	0x261-1:1	BO-OL	1		1	0		read-only (defa-ult)	
Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	Un-it	Acce-ss	PDO Map-ping
Inp-ut 2	0x261-1:2	BO-OL	1		1	0		read-only (defa-ult)	

0x2701 Tuning command

This object is used in tuning mode to specify the parameter that is configured. Altering the default values is recommended for expert users only!

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Tuning command	0x2701:0	UDINT	32			0		read-write	Receive PDO (Outputs)

0x2702 Tuning status

For internal purposes only!

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Tuning status	0x2702:0	UDINT	32			0		read-write	Transmit PDO (Inputs)

0x2703 User MOSI

Reserved for SDK users.

Name	Index:Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Dat-a			ping
User MOSI	0x2703:0	UDINT	32			0		read-write	Receive PDO (Outputs)

0x2704 User MISO

Reserved for SDK users.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
User MISO	0x2704:0	UDINT	32			0		read-write	Transmit PDO (Inputs)

0x2705 Setup wizard completed

Used in OBLAC Drives for marking the successful completion of the setup guide. Intended for internal use only.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Setup wiz-	0x2705:0	USINT	8	0	3	0		read-write	

ard com-pleted									

0x603F Error code

Provides the error code of the last error which occurred. See object `0x203F Error report` for a more detailed description.

Error Code	Meaning
0x2220	Continuous over current (device internal)
0x2250	Short circuit (device internal)
0x2350	Load level fault (I2t, thermal state)
0x2351	Load level warning (I2t, thermal state)
0x3130	Phase failure
0x3131	Phase failure L1
0x3132	Phase failure L2
0x3133	Phase failure L3
0x3210	DC link over-voltage
0x3220	DC link under-voltage
0x3331	Field circuit interrupted
0x4210	Excess temperature device
0x4310	Excess temperature drive

Error Code	Meaning
0x5300	Operating unit
0x6010	Software reset (watchdog)
0x6320	Parameter error
0x7121	Motor blocked
0x7300	Sensor
0x7500	Communication

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Error code	0x603F:0	UINT	16			0		read-only (default)	Transmit PDO (Inputs)

0x6040 Controlword

This object shall indicate the received command controlling the PDS FSA. The bits 7, 3, 2, 1, and 0 shall be supported. The other bits may be supported.

15	11	10	9	8	7	6	4	3	2	1	0
ms		r	oms	h	fr	oms		eo	qs	ev	so

MSB	LSB

LEGEND: ms = manufacturer-specific; r = reserved; oms = operation mode specific; h = halt; fr = fault reset; eo = enable operation; qs = quick stop; ev = enable voltage; so = switch on

Bits of the *controlword*						
Command	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	Transitions
Shutdown	0	X	1	1	0	2, 6, 8
Switch on	0	0	1	1	1	3
Switch on + enable operation	0	1	1	1	1	3 + 4 (NOTE)
Disable voltage	0	X	X	0	X	7, 9, 10, 12
Quick stop	0	X	0	1	X	7, 10, 11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, 16
Fault reset	1	X	X	X	X	15
NOTE Automatic transition to Enable operation state after executing SWITCHED ON state functionality.						

Bits 9, 6, 5, and 4 of the controlword are operation mode specific. The halt function (bit 8) behaviour is operation mode specific. If the bit is 1, the commanded motion shall be interrupted, the PDS shall behave as defined in the halt option code. After releasing the halt function, the commanded motion shall be continued if possible.

The bit 10 is reserved for further use; it shall be set to 0. The bits 11, 12, 13, 14, and 15 are manufacturer-specific.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Controlword	0x6040:0	UINT	16			0		read-write	Receive PDO (Outputs)

0x6041 Statusword

This object shall provide the status of the PDS FSA. The bits 10, 9, and 6 to 0 shall be supported. The other bits may be supported.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ms		oms		ila	tr	rm	ms	w	sod	qs	ve	f	oe	so	rtso
MSB															LSB

LEGEND: ms = manufacturer-specific; oms = operation mode specific; ila = internal limit active; tr = target reached; rm = remote; w = warning; sod = switch on disabled; qs = quick stop; ve = voltage enabled; f = fault; oe = operation enabled; so = switched on; rtso = ready to switch on

State coding

Statusword	PDS FSA state
xxxx xxxx x0xx 0000_b	Not ready to switch on
xxxx xxxx x1xx 0000_b	Switch on disabled
xxxx xxxx x01x 0001_b	Ready to switch on
xxxx xxxx x01x 0011_b	Switched on
xxxx xxxx x01x 0111_b	Operation enabled
xxxx xxxx x00x 0111_b	Quick stop active
xxxx xxxx x0xx 1111_b	Fault reaction active
xxxx xxxx x0xx 1000_b	Fault

If bit 4 (voltage enabled) of the statusword is 1, this indicates that high voltage is applied to the PDS.

If bit 5 (quick stop) of the statusword is 0, this indicates that the PDS is reacting on a quick stop request.

If bit 7 (warning) of the statusword is 1, this indicates the presence of a warning condition. Warning is not an error or fault (examples: temperature limit exceeded, job refused). The status of the PDS FSA is not changed. The cause of the warning is given in the Error report object (203Fh).

If bit 9 (remote) of the statusword is 1, this indicates that the controlword is processed. If it is 0 (local), this indicates that the controlword is not processed. Nevertheless, the PDS may provide actual values, and the PDS may accept COB for configuration data transmission for other parameter objects.

If bit 10 (target reached) of the statusword is 1, this indicates that the PDS has reached the set-point. The set-point is operation mode specific and is defined in detail in the corresponding clauses of this part of the IEC 61800-7 series. Bit 10 is also set to 1, if the operation mode has been changed. The change of a target value by software will alter this bit. If quick stop option code is 5, 6, 7 or 8, bit 10 is set to 1, when the quick stop operation is finished and the PDS is halted. If

halt occurred and the PDS has halted then bit 10 is set to 1, too.

Internal limit active (bit 11) indicates if there is an element that doesn't allow the motion to follow the target value. Several limits exist in the firmware. Max. torque (6072h) limits Torque demand (6074h), Max. motor speed (6080h) limits Velocity demand value (606Bh) and Profile velocity (6081h) in Profile position (PP) mode, Software position limit (607Dh) limits Position demand value (6062h) and position or velocity controller integral limit limits integral part of the PID controller.
Additional limit, Position range limit (607Bh), exists for Position demand value (6062h) in Profile position (PP) mode since demand value can't wrap around the range limit, when profiler is being used.

Operation mode specific bits of Statusword ("oms")

Bits	Cyclic synchronous position control	Cyclic synchronous velocity control	Cyclic synchronous torque control	Profile position mode	Profile velocity mode	Homing mode
bit 13 (0x2000)	Following error bit	not used		Following error bit	not used	Homing error bit
bit 12 (0x1000)	Target position ignored	Target velocity ignored	Target torque ignored	Set-point acknowledge bit	Speed bit	Homing attained bit

Bit 14 and bit 15 are manufacturer-specific.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Status-word	0x60-41:0	UI-NT	16			0		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x605A Quick stop option code

Indicates what action is performed when the quick stop function is executed. The slow down ramp is the deceleration value of the used mode of operations. The table below shows the value def-enitions for the function.

Value	Definition
0	Disable drive function
2	Slow down on quick stop ramp and transit into switch on disabled
6	Slow down on quick stop ramp and stay in quick stop active

Na-me	Index:-Sub	Ty-pe	Bit Si-ze	Options		U-nit	Acce-ss	PDO Map-ping
Qui-ck sto-p opti-on cod-e	0x605-A:0	UI-NT	16	Disable Drive Function	0		read-write	
				Slow down on quick stop ramp and	2			

				transit into switch on disabled			
				Slow down on quick stop ramp and stay in quick stop act-ive	6		

0x6060 Modes of operation

Use this object to request an operation mode.

NOTE This object shows only the value of the requested operation mode, the actual operation mode of the PDS is reflected in Object 0x6061:0.

Value	Definition
-2	Diagnostics mode
-1	Cogging compensation recording mode
+1	Profile position mode
+3	Profile velocity mode
+4	Torque profile mode
+6	Homing mode

Value	Definition
+8	Cyclic sync position mode
+9	Cyclic sync velocity mode
+10	Cyclic sync torque mode

The modes can be switched dynamically while in state `Operation enabled`.

Ensure that all required PDO data is available for the mode you are switching to.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Modes of operation	0x6060:0	SINT	8			0		read-write	Receive PDO (Outputs)

0x6061 Modes of operation display

Displays the actual operational mode.

Value	Definition
-2	Diagnostics mode
-1	Cogging compensation recording mode
0	No mode selected
+1	Profile position mode

Value	Definition
+3	Profile velocity mode
+4	Torque profile mode
+6	Homing mode
+8	Cyclic sync position mode
+9	Cyclic sync velocity mode
+10	Cyclic sync torque mode

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Modes of operation display	0x6061:0	SINT	8			0		read-only (default)	Transmit PDO (Inputs)

0x6062 Position demand value

Provides the value of the position sent to the position controller after position limits are checked. See `0x607D Software position limit`. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Position demand value	0x606-2:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x6064 Position actual value

Provides the actual value of the position encoder. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position actual value	0x606-4:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x6065 Following error window

The range of tolerated position values around the position demand value. If the position actual value is out of the following error window, a following error occurs. A following error may occur when a drive is blocked, an unreachable profile target is commanded, or when the controller is weakly tuned. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it). If the value of the following error window is 0xFFFFFFFF, the following error monitoring will be switched off.

Name	Index:Sub	Type	Bit Size	Min D-	Max D-	Default Data	Unit	Access	PDO Mapping

				at-a	at-a				
Following error window	0x6065:0	UDINT	32			4294967295	Inc	read-write	Receive PDO (Outputs)

0x6066 Following error time out

Configures the time to allow the controller to be outside the error window. After this timeout, bit 13 of the statusword will be set to 1. The units are in milliseconds.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Following error time out	0x6066:0	UINT	16			100	ms	read-write	Receive PDO (Outputs)

0x6067 Position window

Configures the size of the window by reflecting this value on either side of the target position. If the actual value of the position encoder is within this position window, the target position will be regarded as having been reached. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it). If the value of the position window is 0xFFFFFFFF, position window monitoring will be switched off.

Na-	Index-	Typ-	Bit	Mi-	M-	Default	U-	Acce-	PDO

me	:Sub	e	Si-ze	n D-at-a	ax D-at-a	Data	nit	ss	Map-ping
Pos-ition win-dow	0x60-67:0	UDI-NT	32			42949-67295	In-c	read-write	Rece-ive PDO (Out-puts)

0x6068 Position window time

Configures the time required for the position sensor to be within the position window before it's considered to be within the window. The units are in milliseconds.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Pos-ition win-dow time	0x606-8:0	UI-NT	16			0	m-s	read-write	Recei-ve PDO (Out-puts)

0x606B Velocity demand value

Provides the output value of the trajectory generator in velocity control, and the output of the position controller when using the cascaded controller. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-	Ma-x Da-	Def-ault Dat-	U-nit	Acce-ss	PDO Map-ping

				ta	ta	a			
Velocity demand value	0x606B:0	DINT	32			0	rpm	read-only (default)	Transmit PDO (Inputs)

0x606C Velocity actual value

Provides the actual velocity value derived either from the velocity sensor or the position sensor. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity actual value	0x606C:0	DINT	32			0	rpm	read-only (default)	Transmit PDO (Inputs)

0x606D Velocity window

Configures the size of the window by reflecting this value on either side of the target velocity. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it). If the value of the velocity window is 0xFFFF, velocity window monitoring will be switched off.

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Dat-a			ping
Velo-city win-dow	0x606-D:0	UI-NT	16			655-35	rp-m	read-write	Recei-ve PDO (Out-puts)

0x606E Velocity window time

Configures the time the velocity controller needs to be within the velocity window to consider it having reached the target. The units are in milliseconds.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Velo-city win-dow time	0x606-E:0	UI-NT	16			0	m-s	read-write	Recei-ve PDO (Out-puts)

0x606F Velocity threshold

Configures the velocity that is considered zero velocity. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Velo-	0x606-	UI-	16			0	rp-	read-	Recei-

city thres-hold	F:0	NT					m	write	ve PDO (Out-puts)

0x6070 Velocity threshold time

Configures the time required for the actual velocity to rise above the velocity threshold before the motor is considered to be moving. The units are in milliseconds.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Velo-city thres-hold time	0x607-0:0	UI-NT	16			0	m-s	read-write	Recei-ve PDO (Out-puts)

0x6071 Target Torque

Sets the desired torque for the torque controller in profile torque mode. The units are per thousand of rated torque.

Na-me	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Tar-get Tor-que	0x607-1:0	IN-T	16			0		read-write	Recei-ve PDO (Out-puts)

0x6072 Max torque

The configured maximum permissible torque in the motor. The value is given in per thousandth of rated torque.

$$\text{Max Torque} = \frac{\tau_{\text{max_Datasheet}}}{\tau_{\text{rated_Datasheet}}} \times 1000$$

For example, if the rated torque from the datasheet is 200 Nm and maximum torque is 600 Nm,

$$\text{Max Torque} = \frac{600\ \text{Nm}}{200\ \text{Nm}} \times 1000 = 3000$$

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max torque	0x6072:0	UINT	16	0	32767	3000		read-write	Receive PDO (Outputs)

0x6073 Max current

Indicates the configured maximum permissible torque-creating current in the motor. The units are per thousand of rated current.

$$\text{Max Current} = \frac{I_{\text{max_Datasheet}}}{I_{\text{rated_Datasheet}}} \times 1000$$

For example, if the `0x6075 Motor rated current` is 2 Amps and maximum current which can be applied for short duration is 6 Amps,

$$\text{Max Current} = \frac{6\ \text{Amps}}{2\ \text{Amps}} \times 1000 = 3000$$

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Dat-a	Dat-a			ping
Max cur-rent	0x607-3:0	UI-NT	16	0	500-00	300-0		read-write	Recei-ve PDO (Out-puts)

0x6074 Torque demand

Provides the output value of the torque trajectory generator. The units are per thousand of rated torque.

Nam-e	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	U-nit	Acce-ss	PDO Map-ping
Torq-ue dem-and	0x607-4:0	IN-T	16			0		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x6075 Motor rated current

The motor rated current is taken from the motor datasheet. It needs to be configured in rms value. In the datasheet it could be defined in other units (peak value), therefore it needs to be converted into rms value before configuration. The units are in Milliampere [mA]. All relative current data refers to this value (such as `0x6073 Max current`).

Na-me	Index-:Sub	Ty-pe	Bit Si-ze	Mi-n Da-ta	Max Data	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping

Motor rated current	0x6075:0	DINT	32	1	1000000	1	mA	read-write	

0x6076 Motor rated torque

Indicates the nominal torque for the motor. It is taken from the motor's specifications. All relative torque data shall refer to this value (such as `0x6072 Max torque`). For linear motors, the object name is not changed, but the motor rated force value must be entered as multiples of mN (milli Newton). The value shall be given in mNm (milli Newton metre).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Motor rated torque	0x6076:0	DINT	32	1		1	mNm	read-write	

0x6077 Torque actual value

Provides the actual value of the torque, and is computed based on the currents measured in the motor. The units are per thousand of rated torque.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Torque actual value	0x6077:0	INT	16			0		read-only (default)	Transmit PDO (Inputs)

0x6079 DC link circuit voltage

Provides the bus voltage measured at the inverter. The units are in millivolts (mV).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
DC link circuit voltage	0x6079:0	UDINT	32			0	mV	read-only (default)	Receive PDO (Outputs)

0x607A Target position

Sets the desired position for all position modes. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Target pos-	0x607A:0	DINT	32			0	Inc	read-write	Receive PDO

ition									(Out-puts)

0x607B Position range limit

Reports the limits of the position sensor, and is updated when the encoder is reconfigured. On reaching or exceeding these limits, the sensor value will wrap automatically to the other end of the range. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Min position range limit	0x607B:1	DINT	32			2147483648	Inc	read-only (default)	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max position range limit	0x607B:2	DINT	32			2147483647	Inc	read-only (default)	Receive PDO (Outputs)

0x607C Home offset

Defines the difference between the zero position for the application and the machine home position (found during homing). During homing, the machine home position is found and once the homing is completed, the zero position is offset from the home position by adding the home offset to the home position. All subsequent absolute moves are taken relative to this new zero position. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Home offset	0x607C:0	DINT	32			0	Inc	read-write	Receive PDO (Outputs)

0x607D Software position limit

Defines the absolute position limits for the `0x607A Target position` and the `0x6064 Position actual value`. Every new target and actual position are checked against these limits. The limit positions shall be always relative to the machine home position. Limits are disabled by setting them to default values, min. limit to -2147483648 and max. limit to 2147483647. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Min position	0x607D:1	DINT	32			2147483648	Inc	read-write	Receive PDO

limit									(Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max position limit	0x607D:2	DINT	32			2147483647	Inc	read-write	Receive PDO (Outputs)

0x607E Polarity

Indicates if the velocity or position data are multiplied by 1 of by -1, effecitvely flipping the polarity of the command and feedback. The position polarity bit is used for profile position *(pp)* mode and cyclic sync position mode *(csp)*. The velocity polarity bit is used for profile velocity *(pv)* mode and cyclic sync velocity mode *(csv)*.

The valid values of this object are:

`Position polarity flipped` (128) - multiplies the position by -1
`Velocity polarity flipped` (64) - multiplies the velocity by -1
`Both are flipped` (192) - both velocity and position are multiplied by -1

The polarity flag has no influence on the homing mode.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
Polarity	0x607E:0	USINT	8	None: 0 Position: 128		read-write	Receive PDO (Outputs)

				polarity flipped | Velocity polarity flipped | 64 Both flipped | 192				

0x6080 Max motor speed

Sets the max allowed speed for the motor in either direction. It is used to protect the motor and is taken from the motor data sheet. The value shall be given in RPM by default (see `0x60a9 SI unit velocity` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Max motor speed	0x6080:0	UDINT	32	0		1000	rpm	read-write	Receive PDO (Outputs)

0x6081 Profile velocity

Configures the velocity target for the end of the acceleration ramp during profiled motion. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Si-	Min	Max	Default	Unit	Access	PDO Map-

			ze	Da-ta	Da-ta	Dat-a			ping
Pro-file velo-city	0x608-1:0	UDI-NT	32			0	rp-m	read-write	Recei-ve PDO (Out-puts)

0x6083 Profile acceleration

Configures the acceleration used during profiled motion. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Profile accel-eration	0x60-83:0	UDI-NT	32			0	rp-m/-s	read-write	Rece-ive PDO (Out-puts)

0x6084 Profile deceleration

Configures the deceleration used during profiled motion. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-	M-ax D-	Def-ault Dat-	Un-it	Acce-ss	PDO Map-ping

				at-a	at-a	a			
Profile decel-eration	0x60-84:0	UDI-NT	32			0	rp-m/-s	read-write	Rece-ive PDO (Out-puts)

0x6085 Quick stop deceleration

Indicates the deceleration used to stop the motor when the quick stop function is activated. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n D-at-a	M-ax D-at-a	Def-ault Dat-a	Un-it	Acce-ss	PDO Map-ping
Quick stop decel-eration	0x60-85:0	UDI-NT	32	1		100-00	rp-m/-s	read-write	Rece-ive PDO (Out-puts)

0x6086 Motion profile type

Indicates the type of motion profile used to perform profiled motion.

Value	Definition
0	Linear ramp (trapeziodal profile)

Na-	Index:-	Ty-	Bit	Mi-	Ma-	Def-	U-	Acce-	PDO

me	Sub	pe	Si-ze	n Da-ta	x Da-ta	ault Data	nit	ss	Map-ping
Moti-on pro-file type	0x608-6:0	IN-T	16	12-8	0	0		read-only (defa-ult)	Recei-ve PDO (Out-puts)

0x6087 Torque slope

Configures the rate of change of torque. The units are in per thousand of rated torque per second.

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Tor-que slop-e	0x608-7:0	UDI-NT	32			0		read-write	Recei-ve PDO (Out-puts)

0x6088 Torque profile type

Indicates the type of profile used to perform a torque profile.

Value	Definition
0	Linear ramp (trapezoidal profile)

Na-me	Index:-Sub	Ty-pe	Bit Si-ze	Mi-n Da-	Ma-x Da-	Def-ault Data	U-nit	Acce-ss	PDO Map-ping

				ta	ta				
Torque profile type	0x6088:0	INT	16			0		read-only (default)	Receive PDO (Outputs)

0x6091 Gear ratio

Configures the ratio of motor shaft revolutions to driving shaft revolutions. The sensor used for position control is defined to be on the driving shaft. All values are dimensionless.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Motor revolutions	0x6091:1	UDINT	32			1		read-write	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Shaft revolutions	0x6091:2	UDINT	32			1		read-write	Receive PDO (Outputs)

0x6098 Homing method

Name	Index:Sub	Type	Bit Size	Options		Unit	Access	PDO Mapping
Homing method	0x6098:0	SINT	8	[-4] Home on offset (0x2005:-4) before positive endstop	-4		read-write	Receive PDO (Outputs)
				[-3] Home on offset (0x2005:-4) before negative endstop	-3			
				[-2] Home on index pulse before positive endstop	-2			
				[-1] Home on index pulse before negative endstop	-1			

				No homing method assigned: 0 [1] Home on index pulse before positive limit switch rising edge: 1 [2] Home on index pulse before negative limit switch rising edge: 2 [3] Home on index pulse before positive home switch: 3			

				rising edge			
				[4] Home on index pulse after positive home switch rising edge	4		
				[5] Home on index pulse before negative home switch rising edge	5		
				[6] Home on index pulse after negative home switch rising edge	6		

				[7] Home on index pulse before home switch rising edge - Positive movement direction	7		
				[8] Home on index pulse after home switch rising edge - Positive movement direction	8		
				[9] Home on index pulse before home switch fall-	9		

				ing edge - Positive move- ment dir- ection			
				[10] Home on index pulse after home switch fall- ing edge - Positive move- ment dir- ection	10		
				[11] Home on index pulse before home switch rising edge - Negative move- ment dir- ection	11		

				[12] Home on index pulse after home switch rising edge - Negative movement direction	12		
				[13] Home on index pulse before home switch falling edge - Negative movement direction	13		
				[14] Home on index pulse after	14		

				home switch falling edge - Negative movement direction			
				[17] Home on negative limit switch rising edge (switch in inactive state)	17		
				[18] Home on positive limit switch rising edge (switch in inactive state)	18		
				[19] Home on	19		

positive home switch rising edge (switch in inactive state)	
[20] Home on positive home switch rising edge (switch in active State)	20
[21] Home on negative home switch rising edge (switch in inactive state)	21
[22]	22

				Home on negative home switch rising edge (switch in active State)			
				[23] Home on home switch rising edge (switch in inactive state) - Positive move-ment dir-ection	23		
				[24] Home on home switch rising edge (switch in active	24		

				State)- Positive move- ment dir- ection			
				[25] Home on home switch fall- ing edge (switch in active State) - Positive move- ment dir- ection	25		
				[26] Home on home switch fall- ing edge (switch in inactive state) - Positive move- ment dir- ection	26		

				[27] Home on home switch rising edge (switch in inactive state) - Negative move-ment dir-ection	27		
				[28] Home on home switch rising edge (switch in active State) - Negative move-ment dir-ection	28		
				[29] Home on home switch fall-	29		

				ing edge (switch in active State) - Negative move-ment dir-ection			
				[30] Home on home switch fall-ing edge (switch in inactive state) - Negative move-ment dir-ection	30		
				[33] Home on next index pulse - Negative move-ment dir-ection	33		

				[34] Home on next index pulse - Positive move-ment dir-ection: 34 [35] Home on current position (same as mode 37): 35 [37] Home on current position: 37				

0x6099 Homing speeds

Defines the speed at which the motor moves during homing. Two different values can be given; one is used while searching for a limit or home switch, the other is used while searching for zero position (i.e. the index pulse for QEI encoders). The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping

Speed during search for switch	0x6099:1	UDINT	32			0	rpm	read-write	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Speed during search for zero	0x6099:2	UDINT	32			0	rpm	read-write	Receive PDO (Outputs)

0x609A Homing acceleration

Configures the acceleration and deceleration to be used during a homing operation. The units are user-defined (RPM per second by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping

Homing acceleration	0x60-9A:0	UDINT	32			0	rpm/s	read-write	Receive PDO (Outputs)

0x60A9 SI unit velocity

Indicates the user-defined velocity units. The prefix is used by the firmware to scale the velocity; the second byte is fixed to revolutions (0xB4); the third byte can define minutes (0x47) or seconds (0x03).

The supported values are:

- `1 RPM` (value 0x00B44700 or 11814656) - default, but may not be precise enough for some applications.
- `0.1 RPM` (value 0xFFB44700 or 4290004736)
- `0.01 RPM` (value 0xFEB44700 or 4273227520)
- `0.001 RPM` (value 0xFDB44700 or 4256450304)
- `1 RPS` (value 0x00B40300 or 11797248)
- `0.1 RPS` (value 0xFFB40300 or 4289987328)
- `0.01 RPS` (value 0xFEB40300 or 4273210112)
- `0.001 RPS` (value 0xFDB40300 or 4256432896)

Configurations with a gearbox may find `0.001 RPM` or `0.001 RPS` is more appropriate.

Name	Index:Sub	Type	Bit Size	Options	Unit	Access	PDO Mapping
SI unit velocity	0x60-A9:0	UDINT	32	RPM 11814656 0.1 RPM 4290004-736 0.01 4273227-		read-write	

				<table><tr><td>RPM</td><td>520</td></tr><tr><td>0.00-1 RPM</td><td>4256450-304</td></tr><tr><td>RPS</td><td>11797248</td></tr><tr><td>0.1 RPS</td><td>4289987-328</td></tr><tr><td>0.01 RPS</td><td>4273210-112</td></tr><tr><td>0.00-1 RPS</td><td>4256432-896</td></tr></table>			

0x60B1 Velocity offset

Sets a feed-forward velocity as input to the velocity controller. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Velocity offset	0x60B1:0	DINT	32			0	rpm	read-write	Receive PDO (Outputs)

0x60B2 Torque offset

Sets a feed-forward torque as input to the torque controller. The units are in per thousand of rated torque.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Torque offset	0x60B2:0	INT	16			0		read-write	Receive PDO (Outputs)

0x60F4 Following error actual value

Provides the actual value of the position following error. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Following error actual value	0x60F4:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x60FA Control effort

Provides a measure of the control effort (the output of the velocity control loop). The units are per thousand of rated torque.

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Control effort	0x60F-A:0	DINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x60FC Position demand internal value

Provides the output of the trajectory generator in profile position mode. The units are user-defined (increments of the position encoder by default; see `0x6092 Feed constant` to change it).

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Position demand internal value	0x60F-C:0	DINT	32			0	Inc	read-only (default)	Transmit PDO (Inputs)

0x60FD Digital inputs

Bit 0: Shows Negative Limit Switch Status (1 = Active, 0 = Inactive)

Bit 1: Shows Positive Limit Switch Status (1 = Active, 0 = Inactive)

Bit 2: Shows Home Switch Status (1 = Active, 0 = Inactive)

Bit 3: Shows Interlock Status (1 = Drive cannot be enabled, 0 = Drive can be enabled)

Bit 4 to 15: Reserved

The manufacturer-specific bits (16 to 31) show the state of the Drive's digital inputs.

Bit 16: Digital input 1

Bit 17: Digital input 2

:

Bit 31: Digital input 16

For example, if only Digital input 1 is high (and bits 0 to 3 are low), then object value will be 65536 (0x10000)

Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Digital inputs	0x60FD:0	UDINT	32			0		read-only (default)	Transmit PDO (Inputs)

0x60FE Digital outputs

The manufacturer-specific bits allow to set the state of the Drive's digital outputs.

Bit 0: Manual Brake Operation (0=Engage, 1=Disengage)

Bit 1 to 15: Reserved

Bit 16: Digital output 1

Bit 17: Digital output 2

:

Bit 31: Digital output 16

To change the state of an output, the corresponding bit in the **Bit mask** has to be set to 1.

For example, to drive Digtal Output 1 only:

First enable the corresponding mask bit by writting 65536 (0x10000) in **Bit mask**

Then

Writing 65536 (0x10000) in **Physical outputs** will drive the digital output 1 High.

Writing 0 (0x0) in **Physical outputs** will drive the digital output 1 Low.

Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Physical outputs	0x60FE:1	UDINT	32			0		read-write	Receive PDO (Outputs)
Name	Index:Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Bit mask	0x60FE:2	UDINT	32			0		read-write	Receive PDO (Outputs)

0x60FF Target velocity

Sets the target velocity and will be used as input to the trajectory generator. The units are user-defined (RPM by default; see `0x60a9 SI unit velocity` and `0x6092 Feed constant` to change it).

Nam-	Index:-	Ty-	Bit	Mi-	Ma-	Def-	U-	Acce-	PDO

e	Sub	pe	Si-ze	n Da-ta	x Da-ta	ault Dat-a	nit	ss	Map-ping
Tar-get velo-city	0x60F-F:0	DI-NT	32			0	rp-m	read-write	Recei-ve PDO (Out-puts)

0x6502 Supported drive modes

This object indicates the supported drive modes. Each bit equal to 1 within this value indicates a mode which is supported.

Name	Index-:Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	M-ax Da-ta	Def-ault Dat-a	U-nit	Acce-ss	PDO Map-ping
Sup-por-ted drive mode-s	0x65-02:0	UDI-NT	32			197-549		read-only (defa-ult)	Tran-smit PDO (Inpu-ts)

0x6621 Safety statusword

Status of STO and SBC. STO is the 8th bit of the first byte, SBC is the 8th bit of the second byte

Na-me	Index:-Sub	Typ-e	Bit Si-ze	Mi-n Da-ta	Ma-x Da-ta	Def-ault Data	U-nit	Acce-ss	PDO Map-ping

Byte 1	0x6621:1	USINT	8		1	0		read-only (default)	
Name	Index:-Sub	Type	Bit Size	Min Data	Max Data	Default Data	Unit	Access	PDO Mapping
Byte 2	0x6621:2	USINT	8		1	0		read-only (default)	

Communication Area

Index	Name
0x1000	Device type
0x1001	Error register
0x1005	COB-ID SYNC
0x1006	Communication cycle period
0x1008	Manufacturer device name
0x100A	Manufacturer software version
0x100C	Guard time
0x100D	Life time factor
0x1010	Store parameters

	0x1010:1	Save all parameters
0x1011		Restore default parameters
	0x1011:1	Restore all default parameters
	0x1011:2	Restore communication default parameters (disabled)
	0x1011:3	Restore application default parameters (disabled)
	0x1011:4	Restore manufacturer default parameters
0x1016		Consumer heartbeat time
	0x1016:1	Consumer heartbeat time
0x1017		Producer heartbeat time
0x1018		Identity object
	0x1018:1	Vendor-ID
	0x1018:2	Product Code
	0x1018:3	Revision number
	0x1018:4	Serial number
0x1019		Synchronous counter overflow value
0x1023		OS command
	0x1023:1	Command
	0x1023:2	Status
	0x1023:3	Response
0x1024		OS command mode

0x1400		Receive PDO1 parameter
	0x1400:1	COB-ID used by PDO
	0x1400:2	Transmission type
	0x1400:3	Inhibit time
0x1401		Receive PDO2 parameter
	0x1401:1	COB-ID used by PDO
	0x1401:2	Transmission type
	0x1401:3	Inhibit time
0x1402		Receive PDO3 parameter
	0x1402:1	COB-ID used by PDO
	0x1402:2	Transmission type
	0x1402:3	Inhibit time
0x1403		Receive PDO4 parameter
	0x1403:1	COB-ID used by PDO
	0x1403:2	Transmission type
	0x1403:3	Inhibit time
0x1600		Receive PDO1 mapping
	0x1600:1	Controlword
	0x1600:2	Modes of operation
	0x1600:3	Target Torque

	0x1600:4	Target position
	0x1600:5	Target velocity
	0x1600:6	Torque offset
	0x1600:7	Tuning command
0x1601		Receive PDO2 mapping
	0x1601:1	Physical outputs
	0x1601:2	Bit mask
0x1602		Receive PDO3 mapping
	0x1602:1	User MOSI
	0x1602:2	Velocity offset
0x1603		Receive PDO4 mapping
0x1800		Transmit PDO1 parameter
	0x1800:1	COB-ID used by PDO
	0x1800:2	Transmission type
	0x1800:3	Inhibit time
	0x1800:4	reserved
	0x1800:5	Event timer
0x1801		Transmit PDO2 parameter
	0x1801:1	COB-ID used by PDO
	0x1801:2	Transmission type

	0x1801:3	Inhibit time
	0x1801:4	reserved
	0x1801:5	Event timer
0x1802		Transmit PDO3 parameter
	0x1802:1	COB-ID used by PDO
	0x1802:2	Transmission type
	0x1802:3	Inhibit time
	0x1802:4	reserved
	0x1802:5	Event timer
0x1803		Transmit PDO4 parameter
	0x1803:1	COB-ID used by PDO
	0x1803:2	Transmission type
	0x1803:3	Inhibit time
	0x1803:4	reserved
	0x1803:5	Event timer
0x1A00		Transmit PDO1 mapping
	0x1A00:1	Statusword
	0x1A00:2	Modes of operation display
	0x1A00:3	Position actual value
	0x1A00:4	Velocity actual value

	0x1A00:5	Torque actual value
0x1A01		Transmit PDO2 mapping
	0x1A01:1	Analog input 1
	0x1A01:2	Analog input 2
	0x1A01:3	Analog input 3
	0x1A01:4	Analog input 4
	0x1A01:5	Tuning status
0x1A02		Transmit PDO3 mapping
	0x1A02:1	Digital inputs
0x1A03		Transmit PDO4 mapping
	0x1A03:1	User MISO
	0x1A03:2	Timestamp
	0x1A03:3	Position demand internal value
	0x1A03:4	Velocity demand value
	0x1A03:5	Torque demand
0x1C00		Sync Manager
	0x1C00:1	SyncMan 0
	0x1C00:2	SyncMan 1
	0x1C00:3	SyncMan 2
	0x1C00:4	SyncMan 3

0x1C10		SM 0 Assignment
0x1C11		SM 1 Assignment
0x1C12		SM 2 Assignment
	0x1C12:1	SubIndex 001
	0x1C12:2	SubIndex 002
	0x1C12:3	SubIndex 003
	0x1C12:4	SubIndex 004
0x1C13		SM 3 Assignment
	0x1C13:1	SubIndex 001
	0x1C13:2	SubIndex 002
	0x1C13:3	SubIndex 003
	0x1C13:4	SubIndex 004

Manufacturer Specific Area

Index		Name
0x2000		Command object (disabled)
0x2001		Commutation angle offset
0x2002		Position control strategy
0x2003		Motor specific settings
	0x2003:1	Pole pairs
	0x2003:2	Torque constant

	0x2003:3	Phase resistance
	0x2003:4	Phase inductance
	0x2003:5	Motor phases inverted
0x2004		Brake options
	0x2004:1	Pull voltage
	0x2004:2	Hold voltage
	0x2004:3	Pull time
	0x2004:4	Release strategy
	0x2004:5	Controller disable delay
	0x2004:6	DC bus voltage (deprecated)
	0x2004:7	Brake status
	0x2004:8	Minimum displacement (pin brake)
	0x2004:9	Maximum torque (pin brake)
	0x2004:10	Output voltage
	0x2004:11	Switching frequency
0x2006		Protection
	0x2006:1	Undervoltage setpoint
	0x2006:2	Overvoltage setpoint
	0x2006:3	Overcurrent setpoint
0x2008		Cogging torque compensation

	0x2008:1	State
	0x2008:2	Enabled
0x2009		Commutation offset
	0x2009:1	State
	0x2009:2	Applied percent of rated torque
	0x2009:3	Measurement method
	0x2009:4	Phasing controller Kp
	0x2009:5	Phasing controller Ki
	0x2009:6	Phasing controller Kd
0x200A		I2t
	0x200A:1	Enabled
	0x200A:2	Peak time
	0x200A:3	Percent max
	0x200A:4	Protection active
0x200B		Max power
0x2010		Torque controller
	0x2010:1	Controller Kp (disabled)
	0x2010:2	Controller Ki (disabled)
	0x2010:3	Controller Kd (disabled)
	0x2010:4	Field weakening enabled

	0x2010:5	Field weakening percentage
	0x2010:6	Field weakening starting speed
	0x2010:7	Field weakening ending speed
	0x2010:8	Commutation angle measurement delay
	0x2010:9	Switching frequency
	0x2010:10	Settling time
	0x2010:11	Damping ratio
	0x2010:12	Encoder source
0x2011		Velocity controller
	0x2011:1	Controller Kp
	0x2011:2	Controller Ki
	0x2011:3	Controller Kd
	0x2011:4	Controller integral limit
	0x2011:5	Encoder source
0x2012		Position controller
	0x2012:1	Position loop Kp
	0x2012:2	Position loop Ki
	0x2012:3	Position loop Kd
	0x2012:4	Position loop integral limit
	0x2012:5	Velocity loop Kp

	0x2012:6	Velocity loop Ki
	0x2012:7	Velocity loop Kd
	0x2012:8	Velocity loop integral limit
	0x2012:9	Encoder source
0x2013		Gain scheduling
	0x2013:1	Enabled
	0x2013:2	Threshold velocity 0
	0x2013:3	Position loop Kp 0
	0x2013:4	Position loop Ki 0
	0x2013:5	Position loop Kd 0
	0x2013:6	Position loop integral limit 0
	0x2013:7	Velocity loop Kp 0
	0x2013:8	Velocity loop Ki 0
	0x2013:9	Velocity loop Kd 0
	0x2013:10	Velocity loop integral limit 0
	0x2013:11	Threshold velocity 1
	0x2013:12	Position loop Kp 1
	0x2013:13	Position loop Ki 1
	0x2013:14	Position loop Kd 1
	0x2013:15	Position loop integral limit 1

	0x2013:16	Velocity loop Kp 1
	0x2013:17	Velocity loop Ki 1
	0x2013:18	Velocity loop Kd 1
	0x2013:19	Velocity loop integral limit 1
0x2014		Torque window
	0x2014:1	Value
	0x2014:2	Time
0x2015		Velocity feed forward
	0x2015:1	Gain
	0x2015:2	Cutoff frequency
0x2021		Velocity feedback filter
	0x2021:1	Type
	0x2021:2	Cutoff frequency
0x2022		Position feedback filter
	0x2022:1	Type
	0x2022:2	Cutoff frequency
0x2023		Notch filter
	0x2023:1	Enabled
	0x2023:2	Center frequency
	0x2023:3	Rejection band

0x2027		Control input FIR filter
	0x2027:1	Enabled
	0x2027:2	Order
0x2030		Core temperature
	0x2030:1	Measured temperature
0x2031		Drive temperature
	0x2031:1	Measured temperature
0x2038		External scaled measurement
	0x2038:1	Scaled measurement value
	0x2038:2	Analog input
	0x2038:3	Resistance
	0x2038:4	Constant a0
	0x2038:5	Constant a1
	0x2038:6	Constant a2
	0x2038:7	Constant a3
	0x2038:8	Constant a4
	0x2038:9	Constant a5
	0x2038:10	Cutoff frequency
	0x2038:11	Upper error threshold
	0x2038:12	Lower error threshold

0x203F		Error report
	0x203F:1	Description
0x2040		Input counter
0x20E1		High resolution data
	0x20E1:1	Data 1
	0x20E1:2	Data 2
	0x20E1:3	Data 3
	0x20E1:4	Data 4
0x20F0		Timestamp
0x20F2		Assigned name
0x20F3		DC synchronization
	0x20F3:1	DC pulse time
0x2110		Encoder 1 configuration
	0x2110:1	Sensor port
	0x2110:2	Type
	0x2110:3	Resolution
	0x2110:4	Zero velocity threshold
	0x2110:5	Polarity
	0x2110:6	Singleturn offset
	0x2110:7	Access signal type

	0x2110:8	Clock frequency
	0x2110:9	Frame size
	0x2110:10	Multiturn bits
	0x2110:11	Multiturn first bit position
	0x2110:12	Singleturn bits
	0x2110:13	Singleturn first bit position
	0x2110:14	Timeout
	0x2110:15	CRC polynomial
	0x2110:16	Maximum tbusy
	0x2110:17	Status bits active value
	0x2110:18	Parity type
	0x2110:19	First clock delay
	0x2110:20	Data ordering
	0x2110:21	Endianness
	0x2110:22	Index availability
	0x2110:23	Hall sensor port
	0x2110:24	Sinewave cycles per revolution
	0x2110:25	Sinewave resolution
	0x2110:26	Sinusoidal output voltage
	0x2110:27	Filter

0x2111		Encoder 1 feedback
	0x2111:1	Raw position
	0x2111:2	Adjusted position
	0x2111:3	Velocity
0x2112		Encoder 2 configuration
	0x2112:1	Sensor port
	0x2112:2	Type
	0x2112:3	Resolution
	0x2112:4	Zero velocity threshold
	0x2112:5	Polarity
	0x2112:6	Singleturn offset
	0x2112:7	Access signal type
	0x2112:8	Clock frequency
	0x2112:9	Frame size
	0x2112:10	Multiturn bits
	0x2112:11	Multiturn first bit position
	0x2112:12	Singleturn bits
	0x2112:13	Singleturn first bit position
	0x2112:14	Timeout
	0x2112:15	CRC polynomial

	0x2112:16	Maximum tbusy
	0x2112:17	Status bits active value
	0x2112:18	Parity type
	0x2112:19	First clock delay
	0x2112:20	Data ordering
	0x2112:21	Endianness
	0x2112:22	Index availability
	0x2112:23	Hall sensor port
	0x2112:24	Sinewave cycles per revolution
	0x2112:25	Sinewave resolution
	0x2112:26	Sinusoidal output voltage
	0x2112:27	Filter
0x2113		Encoder 2 feedback
	0x2113:1	Raw position
	0x2113:2	Adjusted position
	0x2113:3	Velocity
0x2210		GPIO pin configuration
	0x2210:1	Pin 1
	0x2210:2	Pin 2
	0x2210:3	Pin 3

	0x2210:4	Pin 4
	0x2210:5	Pin 5
	0x2210:6	Pin 6
	0x2210:7	Pin 7
0x2211		GPIO output events
	0x2211:1	Timestamp setpoint
0x2212		GPIO input actions
	0x2212:1	Timestamp on rising edge
0x2213		GPIO position trigger
	0x2213:1	Position setpoint
	0x2213:2	Target encoder
	0x2213:3	Pulse width
0x2214		GPIO global options
	0x2214:1	Voltage level
0x2401		Analog input 1
0x2402		Analog input 2
0x2403		Analog input 3
0x2404		Analog input 4
0x2611		Safety digital input diagnostics
	0x2611:1	Input 1

	0x2611:2	Input 2
0x2701		Tuning command
0x2702		Tuning status
0x2703		User MOSI
0x2704		User MISO
0x2705		Setup wizard completed

Profile Specific Area

Index	Name
0x603F	Error code
0x6040	Controlword
0x6041	Statusword
0x605A	Quick stop option code
0x6060	Modes of operation
0x6061	Modes of operation display
0x6062	Position demand value
0x6064	Position actual value
0x6065	Following error window
0x6066	Following error time out
0x6067	Position window
0x6068	Position window time

0x606B		Velocity demand value
0x606C		Velocity actual value
0x606D		Velocity window
0x606E		Velocity window time
0x606F		Velocity threshold
0x6070		Velocity threshold time
0x6071		Target Torque
0x6072		Max torque
0x6073		Max current
0x6074		Torque demand
0x6075		Motor rated current
0x6076		Motor rated torque
0x6077		Torque actual value
0x6079		DC link circuit voltage
0x607A		Target position
0x607B		Position range limit
	0x607B:1	Min position range limit
	0x607B:2	Max position range limit
0x607C		Home offset
0x607D		Software position limit

	0x607D:1	Min position limit
	0x607D:2	Max position limit
0x607E		Polarity
0x6080		Max motor speed
0x6081		Profile velocity
0x6083		Profile acceleration
0x6084		Profile deceleration
0x6085		Quick stop deceleration
0x6086		Motion profile type
0x6087		Torque slope
0x6088		Torque profile type
0x6091		Gear ratio
	0x6091:1	Motor revolutions
	0x6091:2	Shaft revolutions
0x6098		Homing method
0x6099		Homing speeds
	0x6099:1	Speed during search for switch
	0x6099:2	Speed during search for zero
0x609A		Homing acceleration
0x60A9		SI unit velocity

0x60B1		Velocity offset
0x60B2		Torque offset
0x60F4		Following error actual value
0x60FA		Control effort
0x60FC		Position demand internal value
0x60FD		Digital inputs
0x60FE		Digital outputs
	0x60FE:1	Physical outputs
	0x60FE:2	Bit mask
0x60FF		Target velocity
0x6502		Supported drive modes
0x6621		Safety statusword
	0x6621:1	Byte 1
	0x6621:2	Byte 2

Process Data Objects (PDO)

Process Data Objects (PDO) are used to cyclically transfer information between master and slaves.

There is a standard set of PDO which consists of input objects (TxPDO) and output objects (RxPDO):

- Standard RxPDO mapped Objects
- Standard TxPDO mapped Objects

You can also create your own set of PDO to include the objects which are more suitable for your needs.

- Configuring custom PDO mappings

Standard RxPDO mapped Objects

Index	Name	Bit Length	Bit Offset
0x6040	Controlword	16	RxPDO Mapping 1
0x6060	Modes of operation	8	RxPDO Mapping 1
0x6071	Target Torque	16	RxPDO Mapping 1
0x607A	Target position	32	RxPDO Mapping 1
0x60FF	Target velocity	32	RxPDO Mapping 1
0x60B2	Torque offset	16	RxPDO Mapping 1
0x2701	Tuning command	32	RxPDO Mapping 1
0x60FE:1	Physical outputs	32	RxPDO Mapping 1
0x60FE:2	Bit mask	32	RxPDO Mapping 1
0x2703	User MOSI	32	RxPDO Mapping 1
0x60B1	Velocity offset	32	RxPDO Mapping 1

Standard TxPDO mapped Objects

Index	Name	Bit Length	Bit Offset
0x6041	Statusword	16	TxPDO Mapping 1
0x6061	Modes of operation display	8	TxPDO Mapping 1

© 2023 Synapticon GmbH | Steinbeisstraße 1 | D-71101 Schönaich

0x6064	Position actual value	32	TxPDO Mapping 1
160x606C	Velocity actual value	32	TxPDO Mapping 1
0x6077	Torque actual value	16	TxPDO Mapping 1
0x2401	Analog input 1	16	TxPDO Mapping 1
0x2402	Analog input 2	16	TxPDO Mapping 1
0x2403	Analog input 3	16	TxPDO Mapping 1
0x2404	Analog input 4	16	TxPDO Mapping 1
0x2702	Tuning status	32	TxPDO Mapping 1
0x60FD	Digital inputs	32	TxPDO Mapping 1
0x2704	User MISO	32	TxPDO Mapping 1
0x20F0	Timestamp	32	TxPDO Mapping 1
0x60FC	Position demand internal value	32	TxPDO Mapping 1
0x606B	Velocity demand value	32	TxPDO Mapping 1
0x6074	Torque demand	16	TxPDO Mapping 1

Configuring custom PDO mappings

EtherCAT PDOs are transmitted cyclically in a fixed length. The standard set of PDO can be sent at up to 4000 Hz.

Each PDO mapping can be used for different use cases, for example a diagnostics mapping can be configured in one PDO mapping and this can be discarded for operation.

Mappings can also be added for different drives.

A total of four PDO mappings can be defined with up to 10 objects in each direction.

The defined mappings are configured in objects 0x1A00 - 0x1A03 (TxPDO Mapping) and 0x1600 - 0x1603 (RxPDO Mapping).

The PDO mappings that should be used are defined in the objects 0x1C12 (Incoming: SM Assignment 2) and 0x1C13 (Outgoing: SM Assignment 3).

> Note:
> Since the SM Assignment Mailboxes work as a triple buffer, only 0.3 kByte of data can be used in each PDO mapping.

Changing the content of an object

> Note:
> The content of a mapping object can only be changed in Pre-Operational state.

To change the content of an object:

1. Disable the object by setting its subindex 0 to 0
2. Enter the objects that should be mapped to the mapping entry subindices 1 to n
3. Set subindex 0 to the number n of used entries

> Note:
> If the subindex 0 is not 0 a write access to any other subindex will only be possible if the entered value equals the stored value. This can be used to check the device configuration against the master configuration.

More details can be found in our System Integration Guide

Parameters

Object ID	Name	Description
0x1600	RxPDO Mapping 1	Used for defining the incoming PDO mapping 1
0x1601	RxPDO Mapping 2	Used for defining the incoming PDO mapping 2
0x1602	RxPDO Mapping 3	Used for defining the incoming PDO mapping 3
0x1603	RxPDO Mapping 4	Used for defining the incoming PDO mapping 4
0x1A00	TxPDO Mapping 1	Used for defining the outgoing PDO mapping 1
0x1A01	TxPDO Mapping 2	Used for defining the outgoing PDO mapping 2
0x1A02	TxPDO Mapping 3	Used for defining the outgoing PDO mapping 3
0x1A03	TxPDO Mapping 4	Used for defining the outgoing PDO mapping 4
0x1C00	Sync Manager	This object handles the SM assignment
0x1C10	SM 0 Assign-ment	This object defines the master to slave com-munication (SDO and FoE)
0x1C11	SM 1 Assign-ment	This object defines the slave to master com-munication (SDO and FoE)
0x1C12	SM 2 Assign-ment	This object object defines the incoming PDO map-pings to be used

Object ID	Name	Description
0x1C13	SM 3 Assignment	This object object defines the outgoing PDO mappings to be used

Device Information

- Saving and restoring configurations:
  Configurations can be saved for later use and default configurations can be restored.

These Objects specify how to communicate with a device and how to configure it:

Object ID	Name	Description
0x100A	Manufacturer Software Version	This object shows the Software Version
0x1000	Device Type	This object describes the type of the logical device and its functionality.
0x1008	Manufacturer Device Name	Indicates the name given by Synapticon.
0x1018	Identity	Identification of the EtherCAT device.

Saving and restoring configurations

Overview

This function allows saving and restoring the default configuration or restoring the factory configuration of the drive. The configuration files require a password to avoid accidental changes.

> Note:
> Saving factory defaults is not possible via firmware command. This must be performed offline. In case needed please contact support@synapticon.com for details.

The default values are loaded at startup, first the factory defaults (.factory_config), then the user configuration (config.csv).

- .factory_config contains the original configuration when the drive was shipped.
- config.csv is updated every time the “save to device” button is pressed or the parameters are saved by object 0x1010:1. This stores all parameters at once.

Use case:

Restoring configurations can be used for example for testing parameters without saving and safely going back to the last stable value. Saving the config.csv file can be used:

- as a backup
- for comparing different configurations
- for transferring the configuration to another drive.

The format for factory and user configuration files is CSV, so that it can be modified easily by text editor or spreadsheet. Lines of the config.csv file can simply be copied over into the factory defaults file without modification. The .factory_config file is protected and can not be changed or deleted by regular file operations

Attention:

The config.csv contains motor-individual parameters like the commutation offset or cogging compensation. Moving the config.csv from one drive to the other may require adjusting some parameters individually.

Attention:

The values loaded into the object dictionary are not checked for validity. Ensure that the values are correct when the parameters are set manually, otherwise malfunctions or erratic behavior might be caused that could lead to serious accidents.

When the drive first starts, it locates the .factory_config file and loads it into the object dictionary. If it does not exist, this step is skipped. Secondly, it locates the config.csv and loads the parameters into the object dictionary.

If the config.csv file doesn't exist, the drive will be configured as defined by the .factory_config.

Usage

> Danger:
>
> The drive must be disabled, so that the operation can be carried out safely.

Restore configuration

To restore parameter values from either file into the object dictionary, object 0x1011 (Restore default parameters) has to be triggered. Two subindexes are relevant:

1. Restore all (user) default parameters at subindex 1
2. Restore manufacturer (factory) default parameters at subindex 4

To avoid accidentally restoring the default parameters, restoring is only executed when a password is written to subitem 0x1011:1: 0x64616f6c (daol in Hexcode)

MSB			LSB
d	a	o	l
64h	61h	6Fh	6Ch

> Note:
>
> All parameters that are not stored will be discarded on power off since all parameters are restored automatically at startup.

> Note:
>
> It takes several milliseconds to restore the configuration file.

Save configuration

User defaults - config.csv

To save the state of the object dictionary for loading later, use object Store parameters 0x1010 subitem 1 (Save all parameters).

To protect the file against accidental tampering, a password must be written to the subitem 0x1010:1: 0x65766173 (*evas* in Hexcode)

MSB			LSB
e	v	a	s
65h	76h	61h	73h

After this operation all configuration state parameters are saved to the config.csv file located in the flash memory. It can be recovered via the filesystem using any communication protocol that supports file operations. For details, please refer to Drive operations through File Access over EtherCAT (FoE) .

It's also possible to save the configuration file in OBLAC Drives when clicking on the button "Save parameters".

> Note:
>
> It takes about 3 seconds to save the configuration file.

Limitations

- The drive must be disabled for saving or restoring parameters.
- This feature is supported in all modes of operation.

Parameters

Object ID	Name	Description
0x1010	Store parameters	Triggers storing user-defined parameters to the flash memory of the drive.
0x1011	Restore default parameters	Loads either factory settings or user defined parameters from the drive.

Resources

System Integration Guidelines

Drive Sizing

Choosing a power supply

Wiring guidelines

Earthing

Regenerative energy

Measuring the brake output voltage

Connecting and configuring analog inputs

Temperature sensors

CANopen over EtherCAT CiA 402

Using Statusword and Controlword

Replacing a SOMANET Drive

Tutorials

Tuning guides

Velocity PID Tuning

Position tuning with Gain Scheduling

Cascaded position Tuning

Drive operations through File Access over EtherCAT (FoE)

Handling instructions

About this documentation

System integration guidelines

This is an overview of how to integrate SOMANET servo drives into an application.

Drive Sizing

Choosing a power supply

Wiring guidelines

Earthing

Regenerative energy

Measuring the brake output voltage

Configuring analog inputs

Temperature sensors

CANopen over EtherCAT CiA 402

Using Statusword and Controlword

Replacing a SOMANET Drive

Drive Sizing

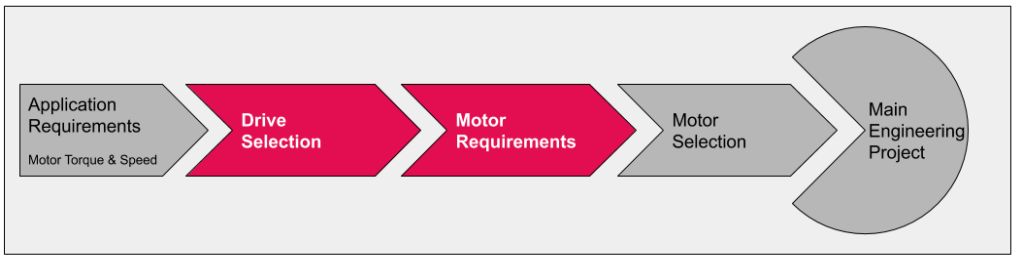
Selecting the correct motor and drive is an important task in any engineering project:

- The system may be damaged if it's not correctly dimensioned.
- With an oversized system, your costs will be too high and your product won't compete.

Only with a properly sized and matching set of motor and drive you can reach your speed and performance targets.

Requirements of the application

The selection highly depends on your application but it's necessary to understand the different parts of the drivetrain:



Application Requirements

Mechanical Power:
$P_{mech} = \omega \cdot \tau$

(Upper limit can be given by gearbox)

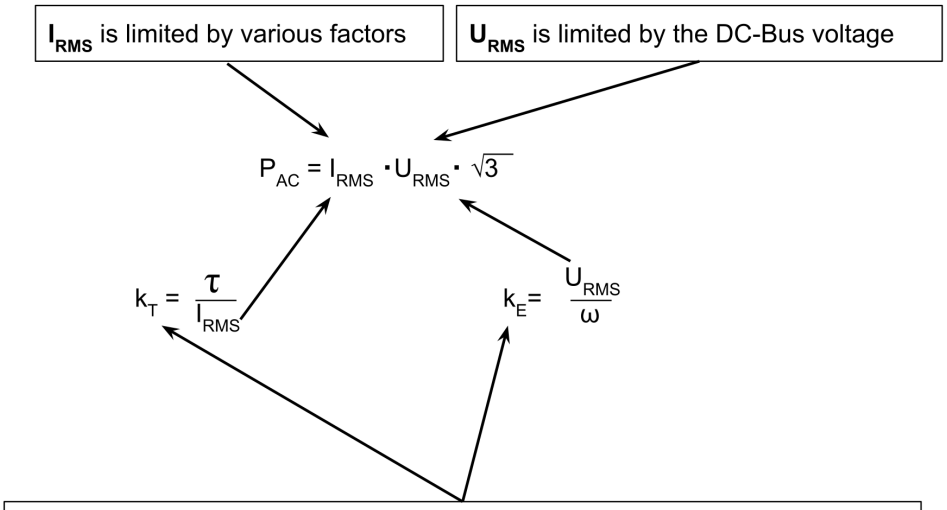
Practically simplified formula:

$P_{mech} = \frac{n}{10} \cdot \tau$

AC Power:

$$P_{AC} = I_{RMS} \cdot U_{RMS} \cdot \sqrt{3}$$

$$U_{RMS} \leq \frac{U_{DC}}{\sqrt{2}} \cdot \eta_{voltage}$$



Motor properties given in Data Sheet:

k$_T$: [Nm/A]
k$_E$: [V/rpm]
$k_T = 1/k_E$
Careful: in the motor's data sheet k_E is often given in V/**k**rpm
The units are depending on temperature: k_T is often measured in a hot state while k_E is measured in a cold state
Therefore, often this equation is not correct!

I_{RMS} is limited	by the Servo Drive	by the Motor	Note
in continuous rated operation	Continuous current	Continuous current	Thermal limit on any component that has to carry current. (FETs and motor windings)
in continuous stall operation	Continuous current	Continuous stall current	Similar thermal limit as in continuous rated operation. Since there are no iron

I_{RMS} is limited	by the Servo Drive	by the Motor	Note
			losses here, the motor can stand a bit more.
in max intermittent operation	Max intermittent current	Max intermittent current	Max current can be applied for a short time only (e.g. 250 ms). Exceeding it would immediately burn the windings.

Servo Drive Consumption

DC Power: $P_{DC} = U_{DC} \cdot I_{DC}$

Calculating the phase current

For selecting a drive, the maximum RMS phase current is the important parameter that should be considered. According to the technical specification table for a given servo drive and a known maximum RMS phase current, an appropriate drive can be chosen for the application. In case the peak current desired for the application is not explicitly known, it can be estimated assuming linear torque characteristics.

$$\text{Maximum peak phase current} = I_{rated} \cdot \frac{\tau_{peak}}{\tau_{rated}} (+\text{nonlinearity buffer})$$

Nonlinearity buffer

For the peak torque, the maximum torque required in the application must be considered - which is not necessarily the peak torque that the motor can provide or its data sheet claims. As real motor characteristics are not linear, a buffer shall be added to the linearly calculated peak current. A buffer of 10% to 50% is reasonable, depending on how much the overload $\frac{\tau_{peak}}{\tau_{rated}}$ is. If the maximum torque required by your application does not exceed the rated torque of your motor (overload <= 1), a buffer of 10% in the equation above is fine. If the peak torque used in your application is 2, 3 or 4 times the rated torque of the motor, the motor can run into its saturation region, where bigger nonlinearity buffers can be required. For details, consult with the motor manufacturer and check the motor curves.

Choosing a power supply

In any application with servo drives, the power consumption and regenerated power must be taken into account. An unstable power line causes harmful harmonics and could damage the system. In most applications, there is more than one drive which requires an accurate sizing of the DC bus. A system with minimum failure rates and little power consumption can be realized by properly sizing the bus capacitors, braking choppers, power supplies, cabling, earthing and using the right connectors. It's recommended to use a protected extra-low voltage supply (PELV) instead of a safety extra-low voltage supply (SELV). If a SELV is used, it's possible that the isolation from the earth can be violated through the servo drive (e.g. through heatsink or mechanical mountings) and the supply becomes PELV.

Supply voltage definitions

The **nominal supply voltage** is the voltage that we recommend to set on your power supply (or use a battery that is within this range).

Maximum voltage ratings divide into **continuous maximum and peak maximum ratings**. The maximum continuous voltage is a voltage that can also be set on your power supply / battery, if all voltage-related effects are well under control. In particular, it must be made sure that neither recuperative braking nor high frequency (typically 32 or 64 kHz) voltage ripple lead to exceeding the allowable maximum continuous voltage value permanently or cyclically. The peak maximum voltage should never be reached in regular operation. It can only be used as a reserve for rare events, no longer than 1 second and not repeatedly.

The **minimum voltage** is for configuration purposes only.

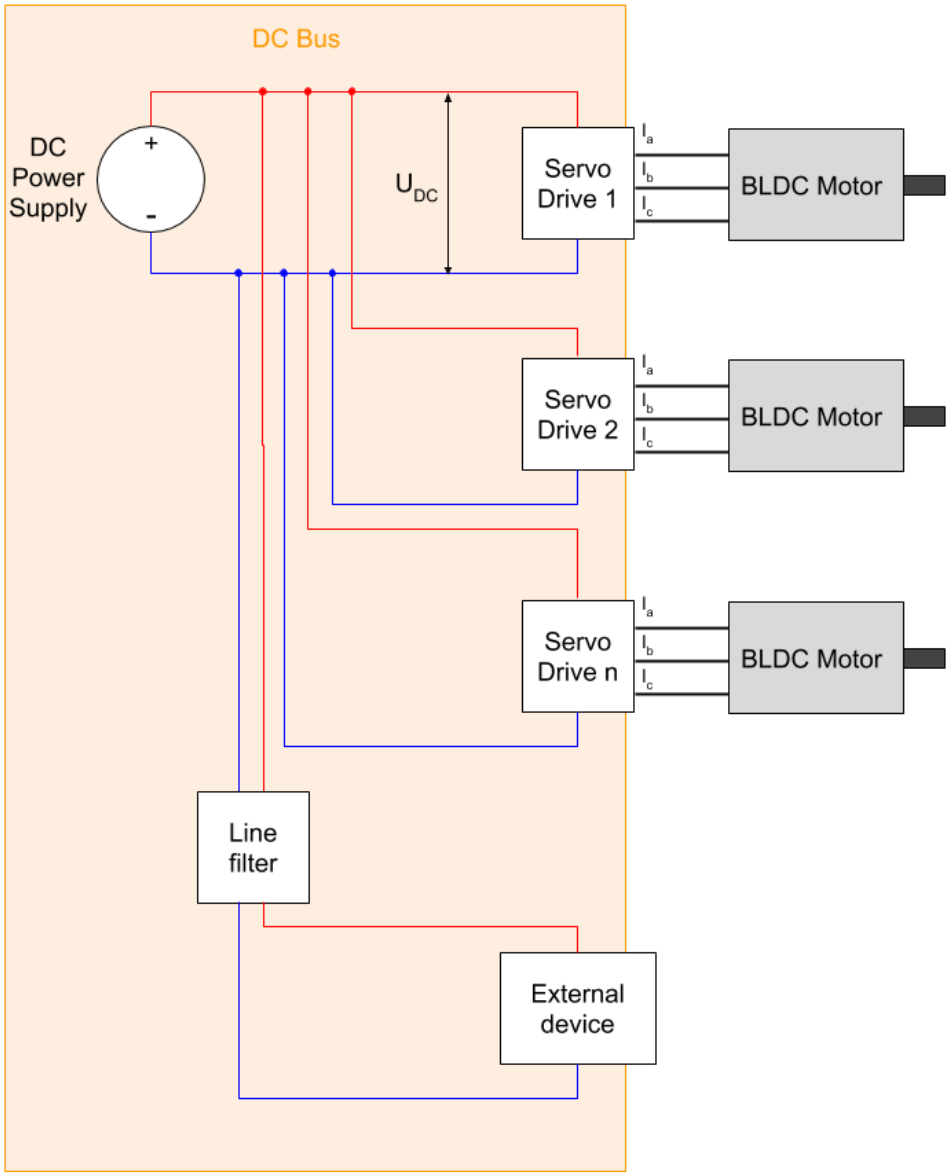
Regenerated voltage

Electrical motors are reversible machines that can act as a motor or generator. When a voltage is applied to the motor, a torque is generated that causes acceleration. But the motor will generate power during deceleration. Regenerative energy can be a serious issue when high inertia, low friction axes undergo fast deceleration. If no counter-measures (particularly adding a braking chopper to the circuit) are taken, the voltage in the DC bus will rise beyond the overvoltage threshold of the servo drives which will cause the drives to switch off and thus the system will become uncontrollable. The problem will be exceeded in multi-axis applications with heavy loads that are affected by gravity. For details, please see section Regenerative energy

Bus capacitance

The bus capacitance consists of all capacitances on a DC link:

DC bus capacitance = power supply capacitance + drives capacitances + other capacitances



An insufficient capacitance will reflect ripples on the DC link that can provide unwanted harmonics which result in more faults. Capacitors can absorb the regenerated power. By using sufficient capacity, the power consumption will be reduced. In addition to coping with ripples on the DC bus, the capacitor can also absorb part of the regenerated energy resulting from deceleration (see Regenerative energy). But using capacitors will not be enough for large amounts of energy - the excessive energy needs to be dissipated with braking resistors.

The amount of energy (in joules) stored by a capacitor is determined by the capacitance (C) and voltage (V) and is given by:

$$E_{cap} = \frac{1}{2} \cdot c(v_2^2 - v_1^2)$$

v_1 is the nominal voltage.

v_2 is the maximum voltage level that is limited by the braking chopper or a braking resistor.

C is the total capacitance of DC BUS (Power supply capacitor + External capacitor + Drive capacitor).

Use case:

- v_1 = 48 V

- v_2 (protection voltage level) = 55 V

- C = 3 mF

E_{cap} calculates to:

$$\frac{1}{2} \cdot 0.003 \cdot (55^2 - 48^2) = 1.08J$$

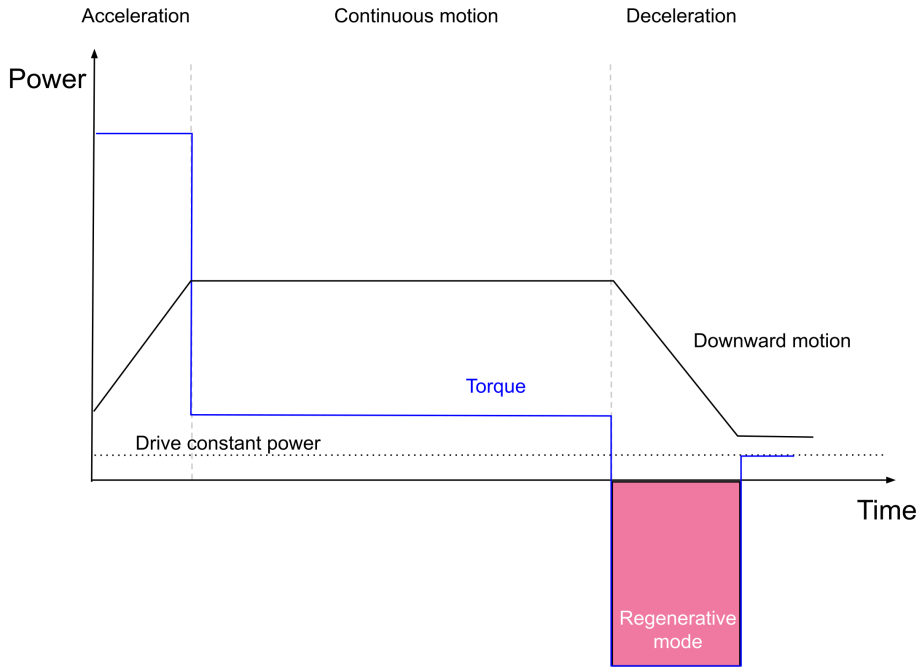
, which means the capacitor can absorb 1.08 J of regenerated energy. Increasing the capacitance is costly and requires space. For estimating the minimum DC link capacitance consider 0.5 µF for 1 W drive power. It is recommended to use a 470 µF capacitor for a SOMANET Node 1000. If the power supply capacitance is more than the estimated capacitance, no additional capacitors are required.

Power supply sizing

For sizing a power supply, the following questions need to be answered:

- How much Watt does the system need?
    - Voltage level and Amperage
    - Power losses and constant power consumption
    - Peak power and average power
- Will regeneration be a concern?
    - Power supply capacity
    - Braking resistor
- At what voltage level should the system operate?
- Should a battery or a switching power supply be used?

Switching power supplies are the most common power supplies for industrial applications, but some applications require batteries. There are some considerations for selecting a suitable power supply. The power supply should be able to provide the demanded power. For the power calculation, the maximum power of each motor and its duty cycle need to be considered for selecting a proper power supply. In rare cases, regenerative power can be helpful when the system features several servo drives which accelerate and decelerate.



Use case:

SCARA robot with four axes, each axis utilises a Node 1000

A 48 V power supply is considered. A 4 kW power supply would be required if all axes need to reach maximum power at the same time. It is not plausible that all the axes require their maximum power at the same time. Considering the robot application, usually two axes can move in a pick and place program at the same time. So in this case a 2 kW power supply could be used. A 20% margin is preferable, so a 2.4 kW power supply is an appropriate power source.

Calculating the amperage

- In a single-axis system, the power supply must be able to provide the peak power required by the application while the real consumption depends on the motors and operating point.

- The above-mentioned rule of thumb can be exceeded by no more than 50% in extreme cases, however in many practical use cases even less power is sufficient. This is particularly the case if high rpm and high torque are not demanded at the same time, but only separately from each other.
- In a multi-axis system with n servo drives, it is typically not required to install n-times the peak power of a single servo drive. This is because in most multi-axis systems, not all axes go to maximum power (maximum torque and maximum speed) at the same time. In robot arms, typically only axes 1 and 2 (SCARA) or axes 2 and 3 (5/6/7 axes robots) are sometimes simultaneously consuming close to full power, so considering the sum of their peak power demands as criteria for the power supply dimensioning is a good starting point.
- Power supply current is derived from the power by

  $P_{DC} = U_{DC} \cdot I_{DC}$

Contactors behind the power supply

SOMANET servo drives are designed for voltages between 12 V and 48 V (60 V Max). Please do not use contactors behind the power supply since the transient-voltage-suppression diodes could get damaged due to the power-up voltage increase (Surge). This is likely to happen when the power-up occurs fast.

This can lead to complete failure of the servo drive.

It is recommended to use a 48 V DC power supply that is common in many industries. A power supply with a significant capacity is required for a servo drive application. Up to 55 V is tolerable for a long time and up to 60 V is acceptable for a time shorter than 1 second. This would allow for regenerative braking of any robot arm, but might be dangerous in systems where gravity provides regenerative power over a longer time, for example in lifting systems or AGVs.

Regenerative currents can occur depending on the commanded trajectory. If the power supply cannot withstand such currents, a braking chopper system must be used additionally.

Wiring Guidelines

For a reliable and robust power interface and communication between the modules, some rules should be followed. A high power, low voltage DC drive needs high current to deliver the maximum power to the motor. Power losses and regenerated energy should also be considered when sizing the DC link cable.

There are a lot of EMI issues around the motor, Drive and power cables. Wires act as antennas, which enables them to pick up radiated signals from nearby sources. The longer the wire, the more pronounced the effect.

Proper grounding, cable shielding, proper connectors and connection are the most critical parts that should be considered for minimum electromagnetic interference (EMI).

DC bus

There are a lot of online sizing tools for selecting the right cable according to a specific application. Some ground rules apply:

- Low resistance cable can increase efficiency.
- Cable flexibility depends on the application.
- Using shielded cable can improve EMC and it will decrease noise.
- Cable length must be as short as possible to avoid noise and copper losses.

EtherCAT cable

It is recommended to use communication cables with a length of no more than 20 meters to minimize the effect of Electromagnetic interference (EMI). Under exceptional circumstances, cable length of up to 100 m can be used.

Preventing connection loss because of connectors or noise, consider high-quality standard cables.

The cables between the drive modules and to the master controller should be as short as possible.

Encoder wiring

Encoder cables should be run in conduits, preferably separate from other wires. If this is not possible, they should be run only with other low-power DC cables.

- Follow the encoder wiring scheme or pinout designated on the encoder datasheet.
- Run cabling in conduits, separated from power cables.
- Shield cables and or individual wires to protect from EMI.
- Avoid ground loops. Ground cables on one side only.
- Consider the shortest cable between drive and encoder.

Earthing

Overview

Earthing is an electrically conductive connection to the earth. Earthing helps to cope with electromagnetic interference (EMI) and should always be considered.

A differentiation can be made between functional earthing (FE) and protective earthing (PE), which is additionally related to electrical safety.

> Note:
>
> In Extra-Low-Voltage devices such as SOMANET Drives, protective earthing is not required but functional earthing is essential to cope with EMI issues.

What is earthing or grounding in an electrical system?

An earthing system (UK and IEC) or grounding system (US) is a common and standardized way to keep the potential of a system to the same level.

Earthing or grounding is the process of transferring the immediate discharge of electricity directly to the earthing plate with low resistance electrical cables or wires. It facilitates a physical connection between the ground and the electrical equipment and ensures the same voltage potential.

Hazards of electricity

Electricity has two hazards which occur when an electric current passes through a person: thermal and shock risk.

Electrical current can be life-threatening above 25 mA in AC and 50 mA in DC. The resistance of the human body is generally between 1 kΩ - 5 kΩ. Assuming a human body has a resistance of 2 kΩ, the dangerous voltage can be calculated as 50 V_{AC} or 120 V_{DC} .

Earthing concepts

Protective Earth

An earthing system (UK and IEC) or grounding system (US) affects the system's safety and the electromagnetic compatibility (EMC).

Protective earthing is a type of earth used to protect people and animals from electric shocks and is effective in the event of a fault.

Protective earthing ensures on one hand that the user is at the same electrical potential as the device housing and on the other hand lets the fuse blow out of a short between phase and ground.

This protects users from electrical shocks when a piece of electric equipment has an insulation failure.

When an insulation failure occurs, the current can pass through the earthing cable path that is more conductive than the human body. The images below show the safe and dangerous conditions with and without PE.

Functional Earth

Functional earth or functional ground is essential to have a trouble-free operation.

Possible consequence when functional earth (FE) is not considered:

- Damage to electronic devices by electrostatic discharge is possible
- EMC issues such as ground loops, random system behavior
- Incorrect reading of the feedback devices that can cause control problems
- Communication problems and loss of data or synchronization.
- Interference to surrounding circuits near the servo drive system.
- Compliance with Electromagnetic Compatibility (EMC) standards and directives may not be given.

Extra Low Voltage systems

SOMANET Drives are in the Extra Low Voltage (ELV) category. ELV voltage range is not life-threatening.

According to the standard for Adjustable speed electrical power drive systems (IEC 61800-5-1:2017), devices with an input voltage lower than 25 V:sub:*AC* and 60 V_{DC} are in the DVCA category.

> Note:
>
> PE connection is not required for the DVCA category.

ELV devices are usually fed by batteries or switching power supplies.

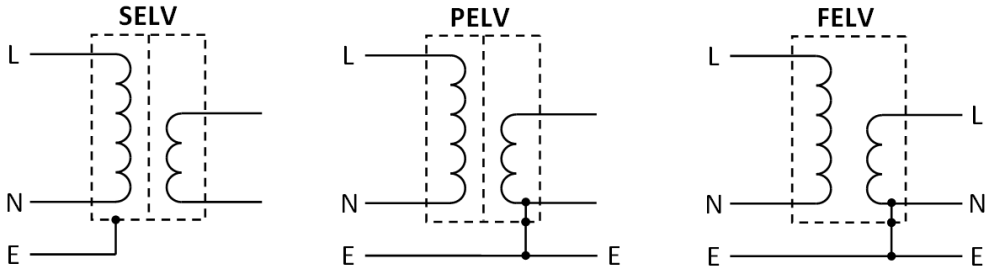
In some cases an earth connection can not be used, for example in mobile robotics.

SELV, PELV, FELV power supplies

Switching power supplies are widely used as a DC source for many applications in the ELV range.

Three categories of switching power supplies can be distinguished that are applicable for servo drives: PELV, SELV or FELV. They differ in their isolation:

- FELV (Functional Extra Low Voltage): No special insulation is required. The load is not isolated by grounding.
- PELV (Protected Extra Low Voltage): Special isolation is required. The load is not isolated by grounding.
- SELV (Separated or Safety Extra Low Voltage): Special isolation is required. The load is isolated by grounding.



PELV is usually sufficient to meet the requirements. SELV is used where extra isolation is required (e.g. swimming pools).

> Attention:
>
> SELV power supply can be violated if the equipment is using a common DC bus.

Earthing connections in SOMANET Products

Table below shows the connection between the heat sink and V- which defines the earth connection.

Product	V- and Chassis	Note
Node 400,1000,2000	Not isolated	V- is connected to the chassis directly.
Braking chopper	Isolated	Because the mounting holes are connected to the V- plane in the PCB, the isolation will break when a conductive Spacer is used between the

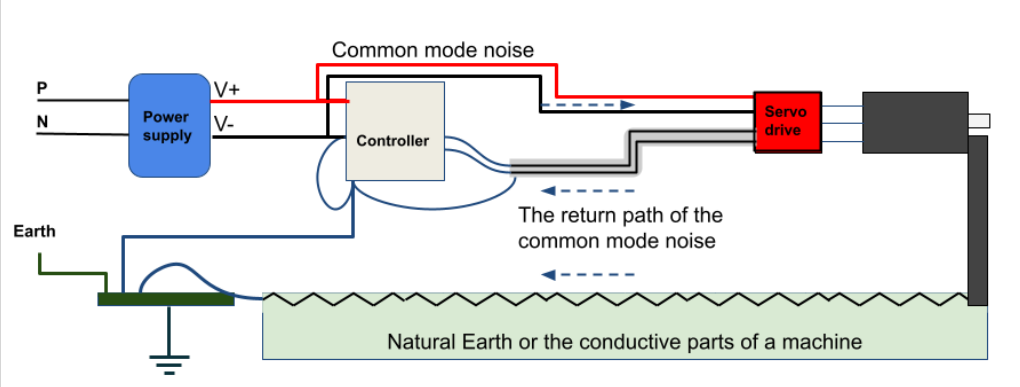
Product	V- and Chassis	Note
		mounting holes on the PCB and chassis. This can violate an existing earthing system and if a SELV power supply is used it may become PELV.

Earthing consideration (guidelines)

The drives generate a large amount of noise because high currents are switched.

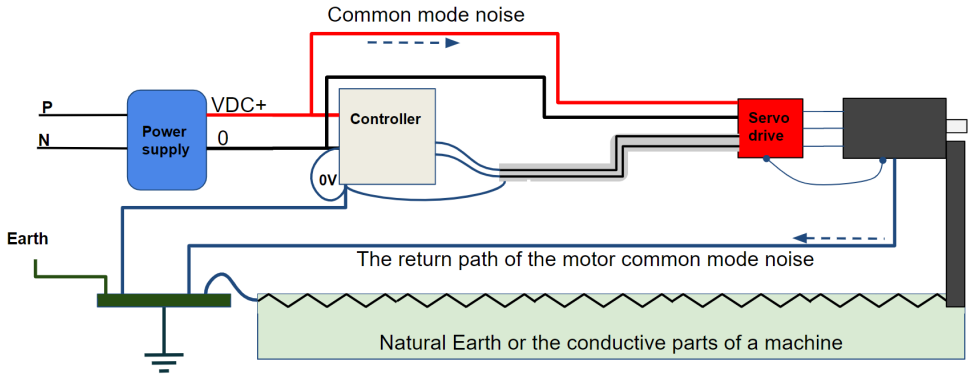
In many use cases, the Motor enclosure is connected to a conductive part of a machine. The parasitic capacitor of the motor is always an alternative pathway which causes common mode noise issues.

The image below shows an improper grounding. In this connection, there is no reliable connection from the motor enclosure to the earth. In this circuit, the noise passes through the shielding and natural earth. The earth impedance on the motor side (power side) is higher than on the control side, which causes a ground loop issue.

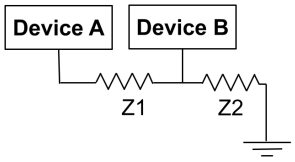


In the next drawing, the motor is connected directly to a low impedance FE with a proper earth connection. In this circuit, the noise passes through the earth cable.

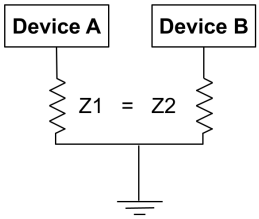
The shielding is connected to the earth correctly, and there is no ground loop because of the unbalanced impedance.



A low impedance earth connection is essential to have less EMI issues. For example, daisy-chaining will increase the EMI issues and ground loops because of the unbalanced ground impedance.



Using a star connection for earthing will result in a better grounding balance and fewer issues with the ground loop and EMI.



Regenerative energy

An electric motor can act as a generator when gravity effects or deceleration drive the motor in generator mode and power is injected into the power line. The amount of regenerated energy is relative to the mechanical energy. Higher deceleration and higher inertia can cause more regenerated power. The regenerated energy can cause overvoltage on the DC link which poses a dangerous situation for all equipment on the shared bus. It is essential to prevent overvoltage for the system to work correctly. The regenerative energy can be either absorbed by a battery (e.g. in an AGV) or burnt by resistors (braking chopper or shunt).

Estimation of regenerated energy

The regenerated energy is related to the mechanical energy. Load, gravity and deceleration are important parameters to define the amount of regenerated energy:

$$E_m = E_k + E_p$$

with:

$$E_m = \text{mechanical energy}$$

$$E_k = \text{kinetic energy}$$

$$E_p = \text{potential energy}$$

To estimate the DC bus capacitor and braking resistor:

1. Calculate the amount of regenerated energy.
2. Calculate the maximum capacitor using the overvoltage protection level.
3. Calculate the dissipated energy.
4. Calculate the amount of power that needs to be dissipated by the braking chopper.

Restoring or dissipating regenerated power

Restoring or dissipating regenerated power is a critical point in many applications where machines feed energy back to the DC bus.
It is easily possible to restore the regenerated power when batteries are used as a power supply or big capacitors can be used. This is an energy-efficient and cost-effective approach, but the component cost may outweigh the benefits.

Otherwise, it is essential to protect the electrical part against overvoltage by dissipating the energy with resistors.
Ideally, the DC bus capacitance can absorb the regenerated power, then no braking resistor or braking chopper would be required in the circuit.

Braking chopper

A braking chopper (or other shunt circuit) is used to burn the energy that can't be recuperated. It is recommended to use SOMANET drives together with one or multiple SOMANET Braking Chopper 48V 500W.

The number of braking choppers depends on the amount of excessive energy (for peak power) and the frequency of regeneration (thermal load).

The SOMANET Braking Chopper is protected against overvoltage and overcurrent. In both cases of overload the chopper will switch off, which will lead to an overvoltage fault on the drive.

Attention:

This can lead to dangerous situations if the robot isn't able to decelerate as necessary.

The calculated number of choppers can be verified in the real system:

1. check the regenerated max power in watching the 3 LEDs on the chopper board:
    - if one or two LEDs are flashing during braking, the number of choppers is sufficient. You have multiple choppers, you can give a try with one less.
    - if 3 LEDs are flashing number you have to check the max voltage with an oscilloscope if voltage is limited (e.g. by the TVR diode of the system)
    - if the drive is having a overvoltage fault, you have to add additional choppers
2. check the thermal behavior of the drive by operating the system in the worst case scenario and check the temperature. If the drive is not switching off at this temperature, the system is well dimensioned.

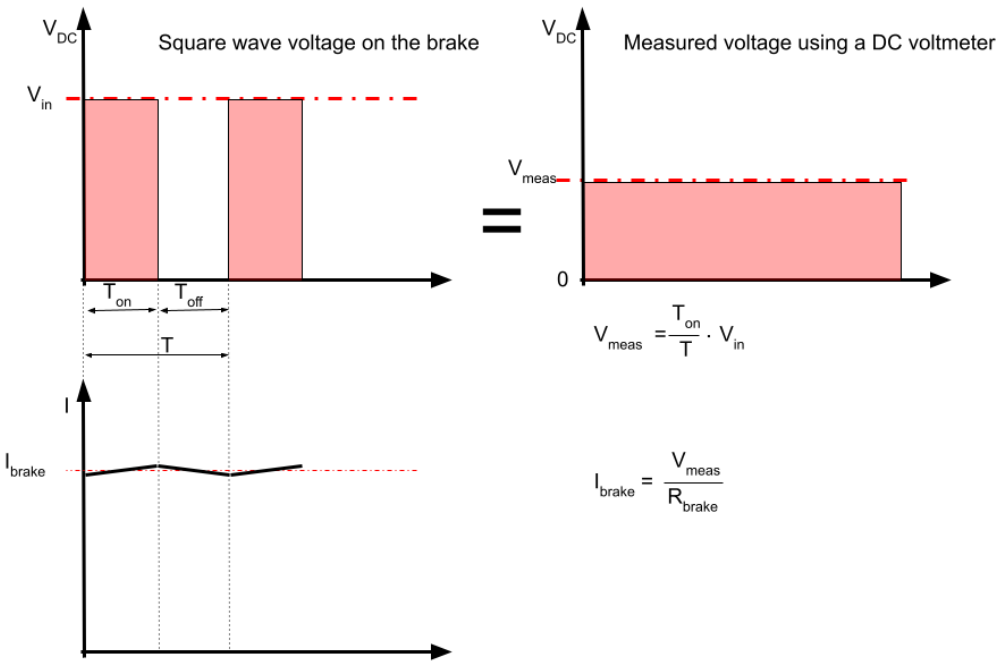
Note:

For optimal performance, connect the braking choppers to the closest possible point to the inverter terminals.

Measuring the brake output voltage

The brake voltage is a square wave signal 16 kHz, 32 kHz or 64 kHz (configurable in subitem 2004:11). A PWM method is used for applying the appropriate voltage to the brake output.

The brake output voltage does not have a negative component. It is connected to an inductor which results in a constant current.



Example

V_{in} (Input DC voltage) = 48 V_{DC}

Brake switching frequency = 16 KHz

Brake hold voltage = 24 V

Brake resistance = 50 Ω

$$T = \frac{1}{\text{Brake switching frequency}} = \frac{1}{16000}$$

$$T_{on} = \frac{\text{Brake hold voltage}}{V_{in}} \cdot T = \frac{24V}{48V} \cdot T = 0.5 \cdot T$$

$$\text{Brake power} = \frac{\text{Brake hold voltage}^2}{\text{Brake resistance}} = \frac{24^2V}{50\Omega} = 11.5W$$

Attention:

Applying more voltage to the brake than required will increase the brake temperature and damage the brake.

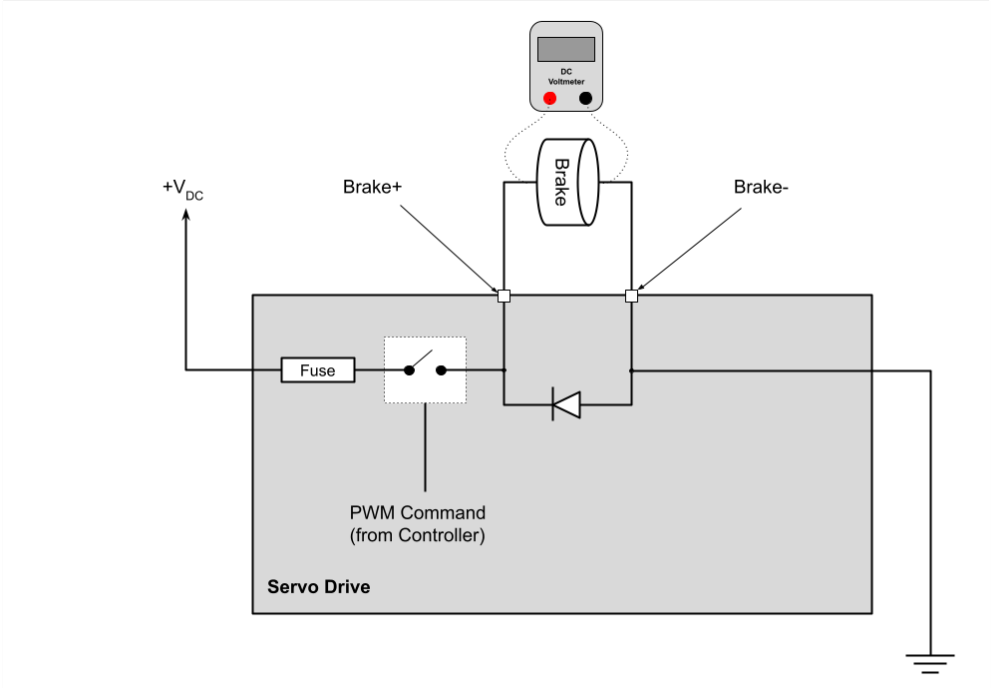
Danger:

If the brake overheats while it's released, the internal deformations might block the brake in the release state!

Measuring the brake current can be done with a DC ampere meter and measuring the brake DC voltage is possible with a common DC voltmeter.

Non-safety brake output

The image below shows a non-safety brake circuit. Measuring the non-safety brake voltage can be done by connecting the voltmeter probe to the brake output (phase D on SOMANET Node) and ground.

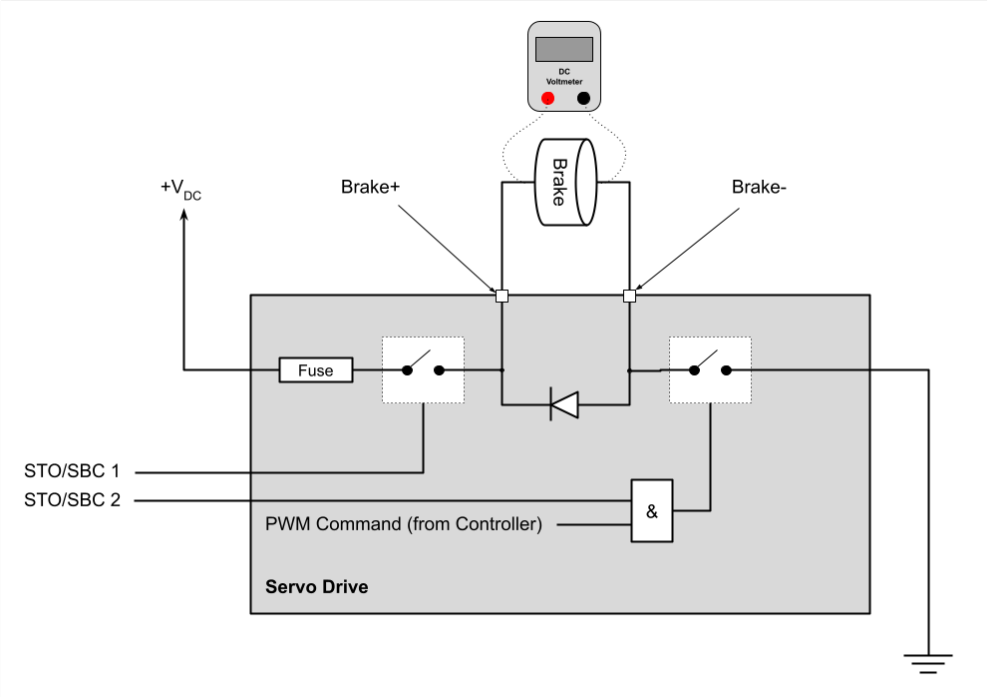


Note:

The brake or a similar electrical load has to be connected to the brake outputs when measuring the output voltage. Without a load, the multimeter will show an incorrect input DC voltage!

Safety brake output

The image below shows the safety brake circuit. The PWM signal is connected to a switch that is connected to Ground (GND). So for controlling the voltage level, it is generating a PWM signal on the GND side.



> Note:
>
> The PWM signal is generated on the Brake- side.

For measuring the safety brake output voltage, B+ and B- must be connected to the voltmeter. Measuring the voltage between B+ and GND or B- and $+V_{In}$ will result in a wrong value.

> Note:
>
> The brake or a similar electrical load has to be connected to the brake outputs when measuring the output voltage. Without a load, the multimeter will show an incorrect input DC voltage.

Configuring analog inputs

Overview

SOMANET Servo drives utilize differential and single-ended analog inputs for measuring a wide range of sensors without adding extra components. Analog sensors can be used for measuring temperature, pressure and torque, both as voltage and resistance depending on their configuration in the firmware.

With an attached voltage division circuit and a polynomial function in the software, the drive carries out all calculations. The calculated temperature, pressure or torque value will be submitted to the master by SDO/PDO for protection or display.

The sensor configuration depends on the used type of sensor as described in this section.

Measuring analog input voltages

SOMANET Servo drives have different inputs to measure analog values. They are connected to an integrated ADC (Analog to Digital converter) with 12-bit resolution (0-4095 ticks).

Please refer to the installation manual for the Analog In specifications for SOMANET Node.

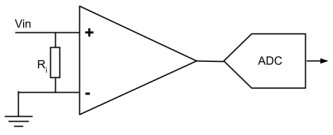
Analog inputs must be used (Single-Ended or Differential) for measuring analog sensors or any kind of analog voltage.

Analog sensors provide a resistance, current or voltage value and are available for pressure, temperature, torque and many other applications.

For a basic understanding of the analog inputs and signal connection concepts the terms V_{diff}, V_{in} and V_{cm} need to be explained.

Single-ended input

Single-ended inputs measure the voltage between the input signal and ground.



The input voltage needs to be scaled to the ADC ticks and the allowed voltage range.

Single-ended analog inputs are highly susceptible to electromagnetic noise.

Relevant parameters for single-ended inputs are V_{in}, $V_{in,lim}$ and R_i.

$$\mathrm{ADC(ticks)} = \frac{V_{in}}{V_{in,max}} \cdot 4095 \text{ if } 0 \leq V_{in} \leq V_{in,max}$$

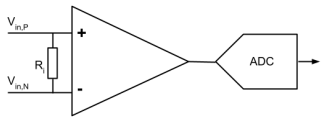
ADC(ticks) = 0 if $V_{in} < 0$

ADC(ticks) = 4095 if $V_{in} \geq V_{in,max}$ (Saturation)

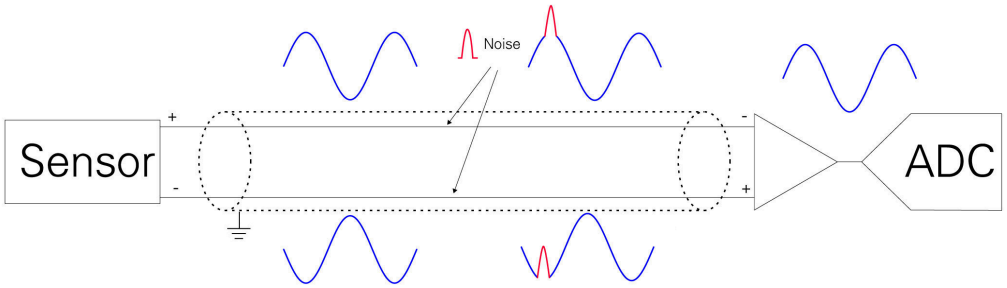
V_{in} must be within a range of $V_{in,min}$ and $V_{in,max}$, otherwise the device will be damaged.

Differential input

The differential inputs measure the voltage between two distinct inputs.



The signals are multiplied by -1 in the source side and compared by the drain to counteract the effect of common mode noise. The advantages of a differential input is its wider input range, the possibility to measure negative voltage and the robustness against common mode noise. Therefore, they show better results in environments which are exposed to electromagnetic disturbances.



For differential inputs the values for V_{diff}, $V_{in,P}$, $V_{in,N}$ and V_{cm} are important, .

$V_{diff} = V_{in,P} - V_{in,N}$

$$V_{cm} = \frac{V_{in,P}+V_{in,N}}{2}$$

$V_{in,P}$ and $V_{in,N}$ are the voltages to GND

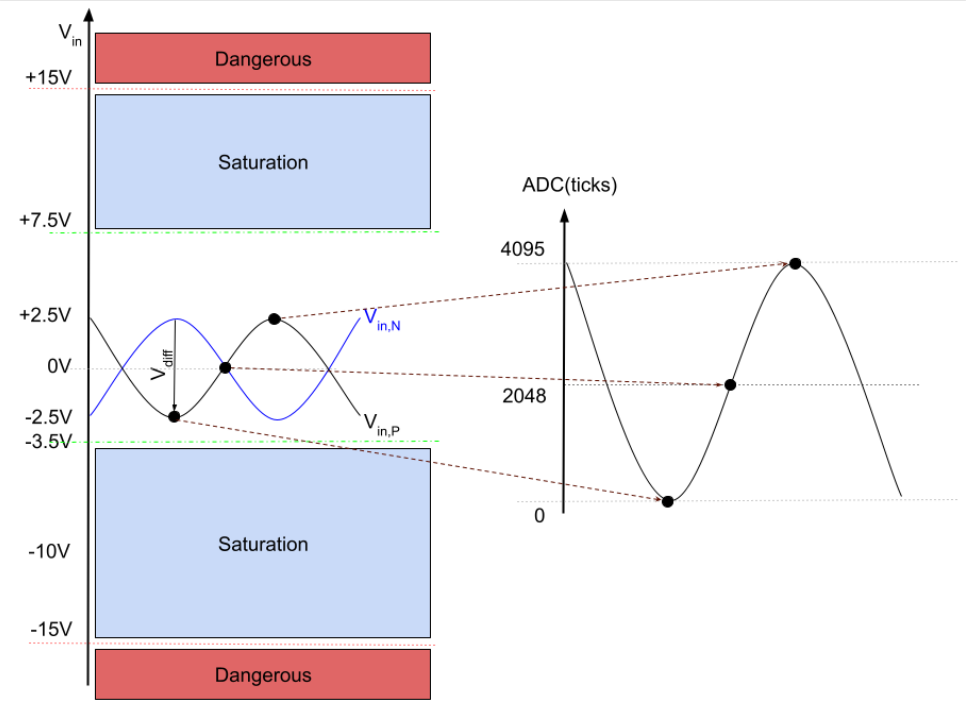
The ADC ticks are calculated by this formula:

$$ADC[ticks] = \frac{V_{diff}}{V_{diff,max}-V_{diff,min}}$$

if $V_{diff,min} \leq V_{diff} \leq V_{diff,max}$

ADC [ticks] = 0 if $V_{diff} < V_{diff,min}$

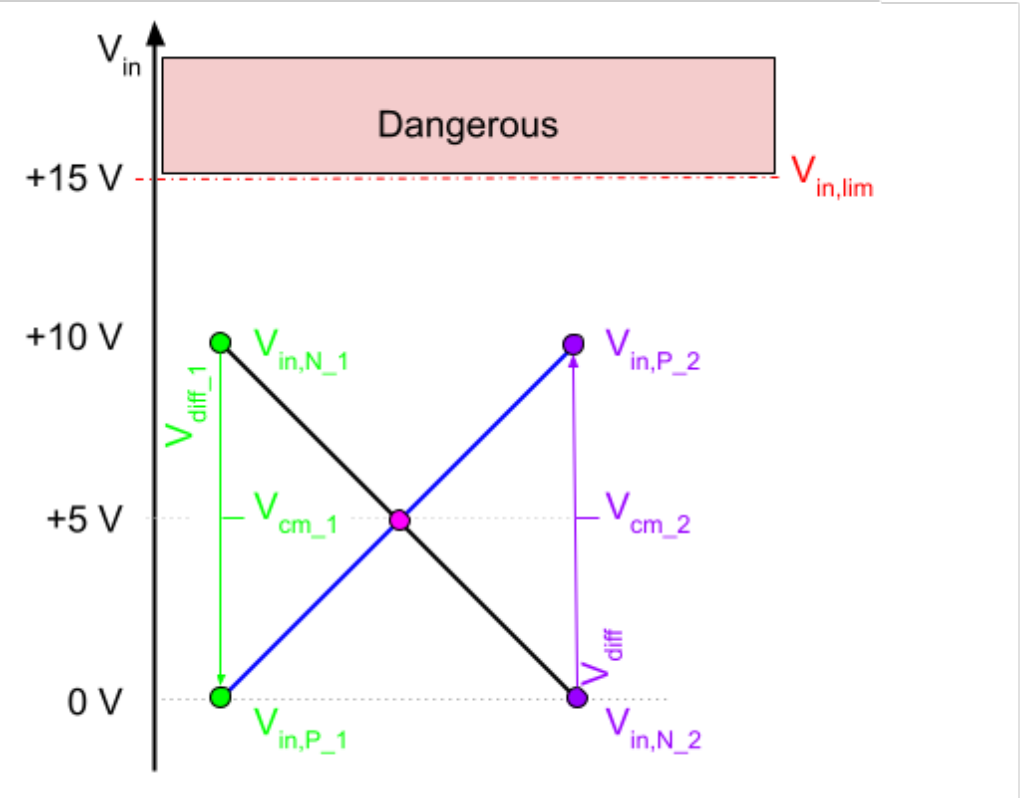
ADC(ticks) = 4095 if $V_{diff} > V_{diff,max}$ (Saturation)



$V_{in,P}$ and $V_{in,N}$ must be within a range of $V_{lim,min}$ and $V_{lim,max}$, otherwise the device will be damaged.

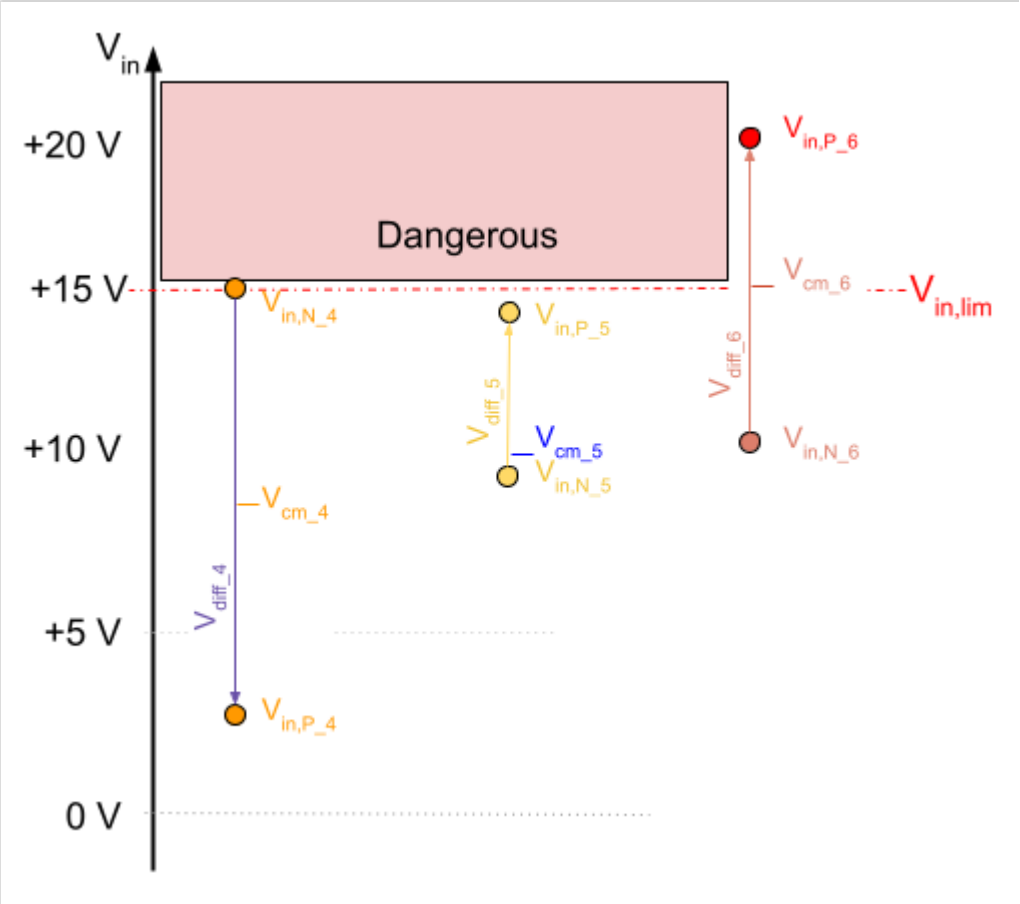
As a kind of reference voltage V_{cm} is used to avoid non-linearity of the input. V_{cm} must be within a range of $V_{cm,min}$ and $V_{cm,max}$, otherwise the measured values will have non-linearity faults.

Example for normal values (On SOMANET Node)



Example	Vin,P	Vin,N	Vdiff	Vcm	ADC (ticks)
1	0	+10	-10	+5	0
2	+10	0	+10	+5	4095
3	0	0	0	0	2047

Examples of faulty values



Example	Vin,P	Vin,N	Vdiff	Vcm	ADC (ticks)
4	+3	+15		+9	
5	14	9	5		
6	20	10	10	15	Damage!

Temperature Sensor input

In the Circulo products family, a dedicated temperature port is available which is directly connected to the ADC input and has a high input resistance. On this port the resistance of a temperature sensor can be measured with high accuracy.

> Note:
>
> It is recommended to use the temperature input port only for measuring temperature with an **RTD** temperature sensor.

The signals are available in different objects, for details, please refer to the installation manual of SOMANET Node.

For details about using temperature sensors please check our System integration guide on temperature sensors.

Internal Resistance

SOMANET Node and Circulo servo drives support different types of analog inputs. These inputs support a wide range of industrial sensors, e.g. torque, temperature, position and speed but have different internal resistance, saturation values and voltage limits.

Temperature sensors

One of the most important parameters for protecting a motor against overheating is the motor's temperature. The motor temperature should be monitored to prevent over-temperature and damage to the motor winding.

Types of temperature sensors

Several types of sensors are well known and widely used in the industry:

- Thermocouple
- Thermistors
- Resistance temperature detectors (RTDs)

Thermocouple

A thermocouple is an electrical device consisting of two dissimilar electrical conductors forming an electrical junction. A thermocouple produces a temperature-dependent voltage as a result of the thermoelectric effect.

> Note:
>
> The generated voltage by a thermocouple is not enough to be measured by analog inputs and **require an interface circuit** (transducers) to increase the output voltage level.

Thermistors

PTC and NTC thermistors can be used as well, but NTC thermistors have the **disadvantage that a wire brake can be confused with low temperatures.**

Resistance temperature detectors

Resistance temperature detectors (RTDs) are manufactured from metals whose resistance increases with temperature.

SOMANET servo drives can directly measure the voltage or resistance so RTDs like KTY84-130 and PT1000 can be directly attached.

A wide range of RTD sensors can be measured by the temperature sensor input (SOMANET Circulo only) or analog inputs.

> Note:
>
> This application note is focused on RTDs.

Pull-up resistor selection

For measuring the temperature using an RTD, the sensor's voltage needs to be adjusted to a specific temperature range by adding a pull-up resistor.

The analog input can be configured to read the input voltage or input resistance. Subitem 0x2038:3 defines the value of the pull-up resistor. Two modes for measuring the analog input value can be distinguished:

Pull-up resistor (0x2038:3) = 0

The measured value will be the exact value of the analog input voltage.

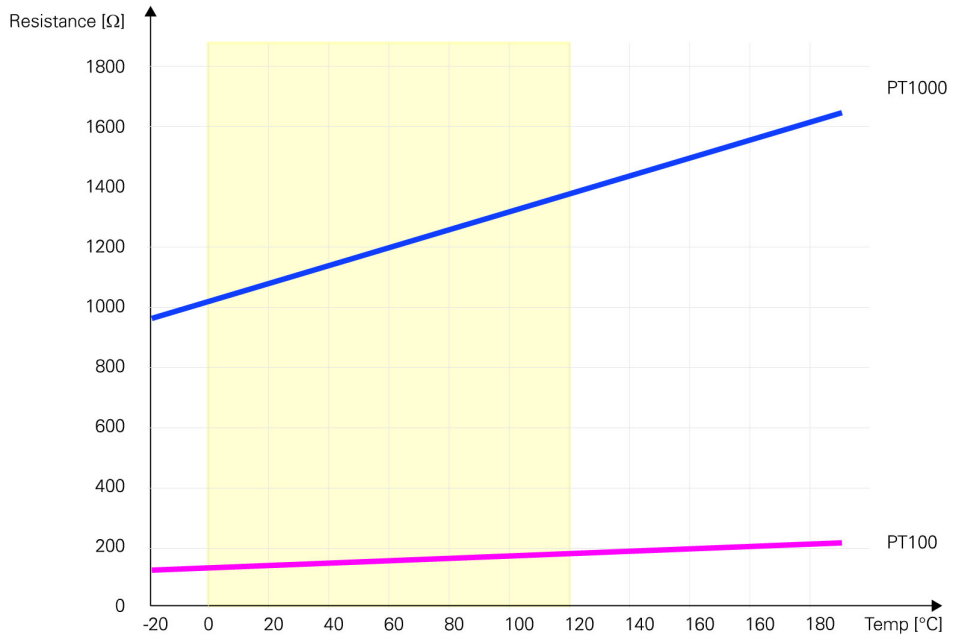
0 = 0 volts and 4096 = full scale (according to the full scale voltage of the analog input)

Pull-up resistor (0x2038:3) > 0

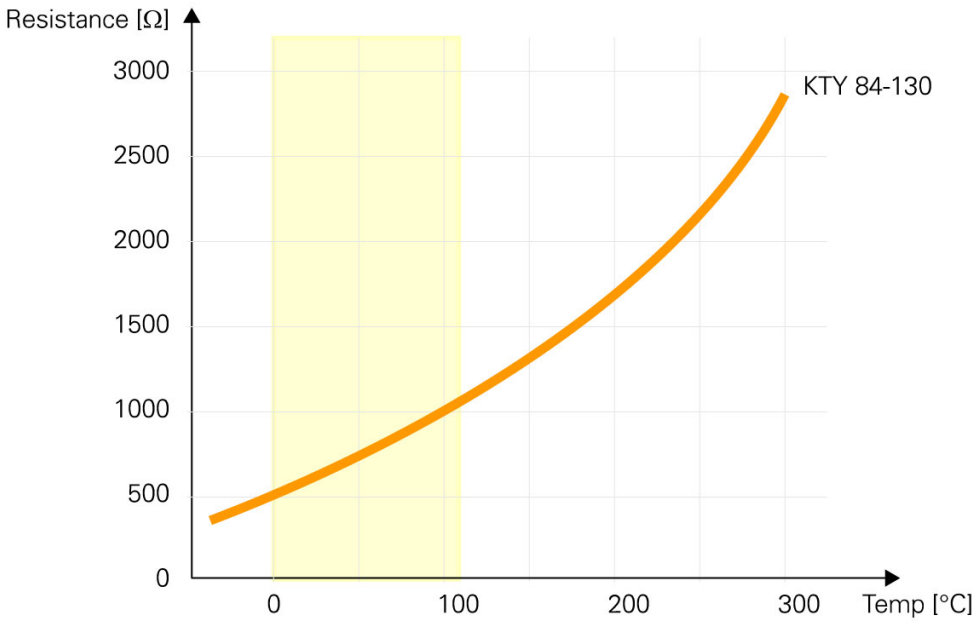
In this mode, the firmware will calculate the resistor value (in Ω) connected to the input terminals. This feature is useful when using RTD sensors or another resistive sensor.

> Note:
>
> RTDs with a minimum of 3 Ω resistance change per °C result in an acceptable accuracy.

According to the curves below, PT1000 and KTY-130 are good fitting sensors, both sensors are almost linear in the desired range.



→ PT1000: 3.9 Ω per °C



→ KYT 84-130: 5 Ω per °C

Note:

Some analog inputs have a low input resistance. The internal input resistance is connected in parallel to the input.

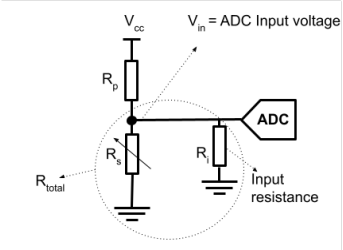
Use case:

Example: Measuring a motor temperature using a PT1000 temperature sensor

According to the PT1000 resistance table (provided, for example, from your sensor's data sheet) the resistance of the sensor should be 1000 Ω for 0° C and 1385 Ω for 100° C.

The reference voltage is 5 V, and the ADC resolution is 12 bits, so the ADC output value would be 4095 for full scale input voltage.

The image below depicts a schematic of input circuits. The ADC will measure the input voltage which is proportional to the resistance.



The sensor resistance and the input resistance are in parallel which means:

$$R_{total} = \frac{R_i * R_s}{R_i + R_s}$$

If the temperature sensor is used on the single-ended analog input of Circulo then $R_s = R_{total}$ because $R_i >> R_s$.

$$\text{The input voltage is} = 5V * \frac{R_{total}}{R_p + R_{total}}$$

To evaluate the accuracy, the amount of ticks for the measured temperature range needs to be considered: Temperature range / Amount of ticks for the temperature range = Temperature measured / tick

Use case:

Example: ADC output on Circulo for a PT1000 temperature sensor

R_p = 4.7 kΩ, R_i be neglected because $R_i >> R_s$

$$\text{ADC input voltage for } 0^\circ = 5\ \text{V} * \frac{1000\Omega}{4700\Omega + 1000\Omega} = 0.8772\ \text{V}$$

$$\text{ADC input voltage for } 100^\circ = 5\ \text{V} * \frac{1385\Omega}{4700\Omega + 1385\Omega} = 1.1382\ \text{V}$$

$$\text{The converted value (ADC ticks) for } 0^\circ \text{ is} = 4095 * \frac{0.8772V}{5V} = 718$$

$$\text{The converted value (ADC ticks) for } 100^\circ \text{ is} = 4095 * \frac{1.138V}{5V} = 932$$

Temperature range in ticks: 932 - 718 = 214 ticks for 100° C.

This leads to an accuracy of $\frac{100^{\circ}C}{214} = 0.5^{\circ}$ for measuring the temperature

Now the accuracy of a PT100 with an R_p of 4.7 kΩ can be calculated for comparison:

Use case:

Example: ADC output on Circulo for a PT100 temperature sensor.

$$\text{ADC input voltage for } 0^{\circ} = 55\ \text{V} * \frac{100\Omega}{4700\Omega+100\Omega} = 0.1042V$$

$$\text{ADC input voltage for } 100^{\circ} = 5\ \text{V} * \frac{139.16\Omega}{4700\Omega+485\Omega} = 0.134V$$

$$\text{The converted value (ADC ticks) for } 0^{\circ} = 4095 * \frac{0.1042V}{5V} = 85$$

$$\text{The converted value (ADC ticks) for } 100^{\circ} = 4095 * \frac{0.468V}{5V} = 110$$

$$\text{Temperature range in ticks: } 110 - 85 = 25 \text{ for } 100^{\circ}\text{C}$$

$$\text{Accuracy: } \frac{100^{\circ}C}{25} = 4^{\circ} \text{for measuring the temperature}$$

Polynomial or Quintic function

After converting the analog input to digital, a five order polynomial function can be used to compensate the non-linearity of the input.

There are two methods for compensating the errors:

- Using the master controller and perform the calculation on the master side.
- Using the integrated polynomial function

$$F(x) = a_5 * x^5 + a_4 * x^4 + a_3 * x^3 + a_2 * x^2 + a_1 * x + a_0$$

> Note:
>
> The polynomial can only be used for one of the analog inputs.

Two methods are available:

- Input voltage measurement
- Input resistance measurement

Case 1: Input voltage measurement

The polynomial function can be used to compensate the non-linearity of the input voltage:

$$y(V_{in}) = a_5 * V_{in}^5 + a_4 * V_{in}^4 + a_3 * V_{in}^3 + a_2 * V_{in}^2 + a_1 * V_{in} + a_0$$

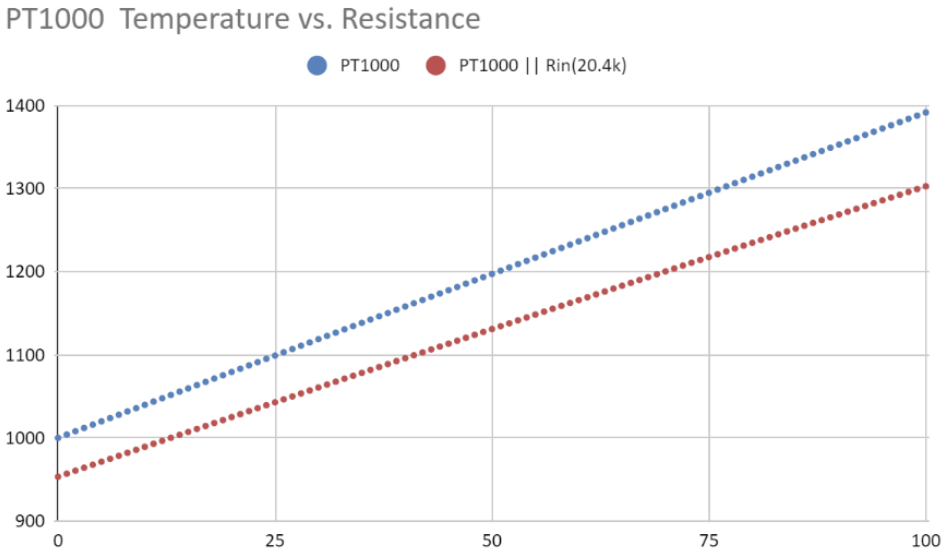
Case 2: Input resistance measurement

When an RTD is used, the temperature is a function of the measured resistance:

$$y(R) = a_5 * R^5 + a_4 * R^4 + a_3 * R^3 + a_2 * R^2 + a_1 * R + a_0$$

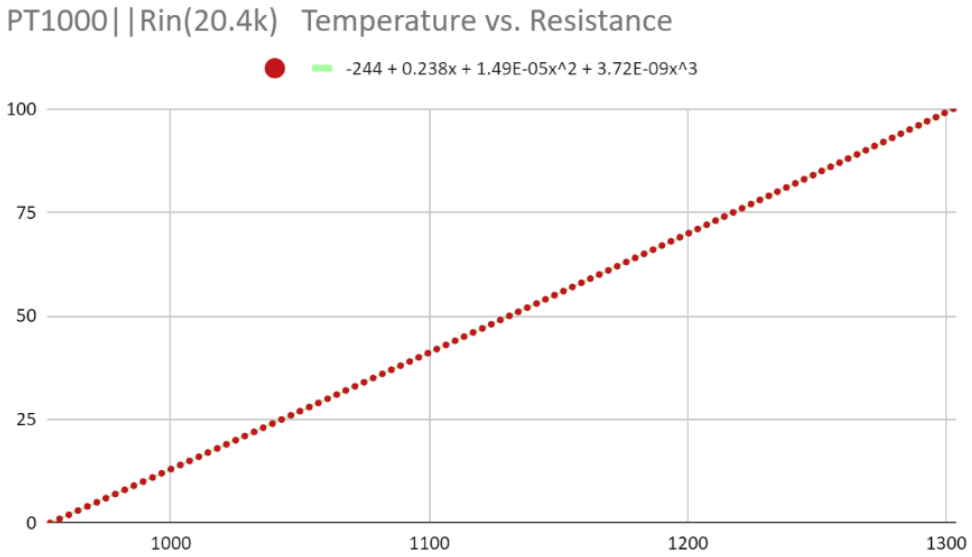
If the internal analog input is chosen with a pull-up resistor, the result of the calculation would be the resistor's amount in Ω connected to the input terminals. With the coefficients set as $a_0 = 0$, $a_1 = 1$, $a_2 = 0$, $a_3 = 0$, $a_4 = 0$, $a_5 = 0$ the result equals the exact value of the sensor resistance.

The image below shows the PT1000 curve with and without input resistor.



In the curves below the polynomial coefficients are given as calculated in EXCEL.

SOMANET Node has an input resistance of 20.4 kΩ (for the single-ended analog input). This resistance is in parallel with the temperature sensor (PT1000). So the curve has to be redrawn with parallel input resistance in order to calculate the polynomial function coefficients.

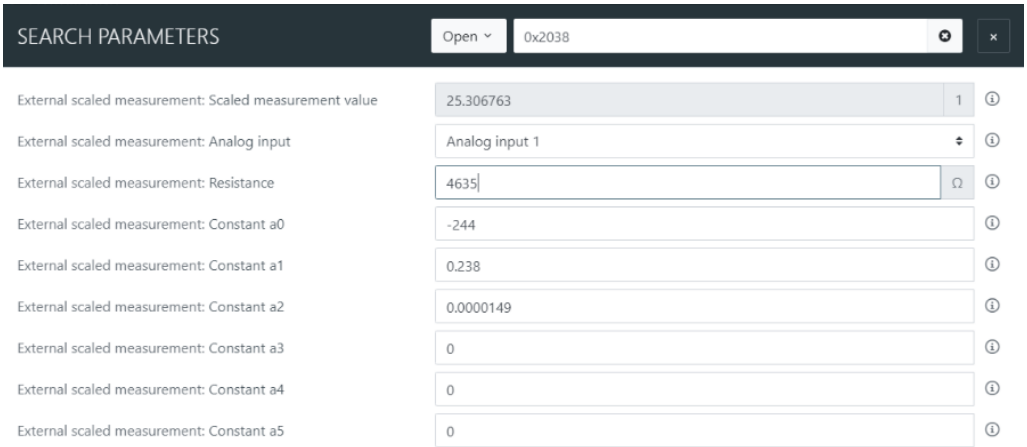


The equation below is related to the PT1000 temperature curve vs. PT1000 resistance|| Input resistance .

Temperature = -244 + 0.238x + 1.49E-05x² + 3.72E-09x³

=> a_0 = -244, a_1 = 0.238, a_2 = 1.49E-05

The image shows how to add the coefficient in Node.

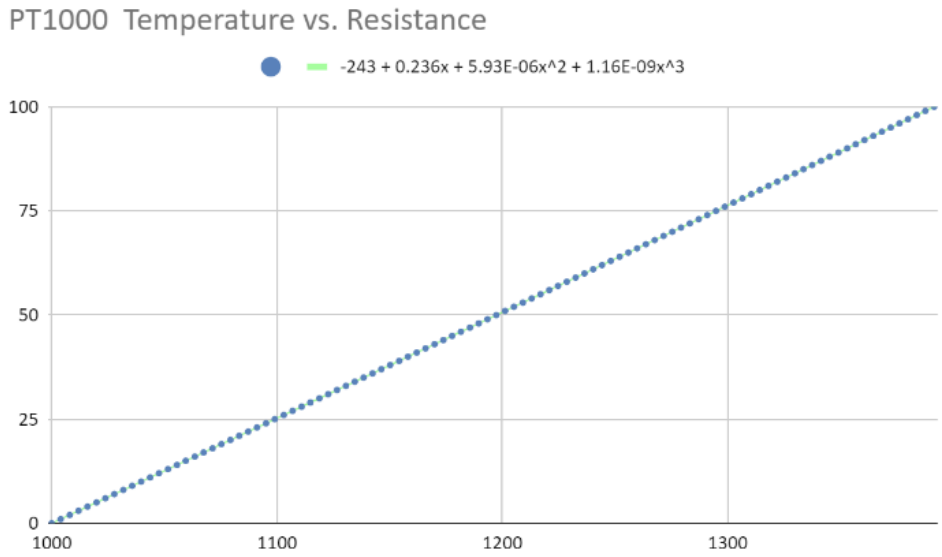


The accuracy of the measured temperature is related to the accuracy of all elements. For example, a 4.7 kΩ resistor is set as the pull-up resistor but the measured value using a precise LCR meter is 4635 Ω.

The image below shows the resistance vs. temperature curve for a PT1000 sensor.

Temperature = -243 + 0.236x + 5.93E-06x² + 1.16E-09x³

=> a_0 = -243, a_1 = 0.236, a_2 = 5.93E-06



If a higher sensitivity in a particular range of temperatures is required, the polynomial can be calculated only for that range of temperatures.

> Note:
>
> For defining the protection level, one point is enough and the polynomial function is not required.

For example, the resistance value can be calculated for 90° in Node:

The resistance of a PT1000 is 1353 for 90° C, the input resistance of Node is 20.4 kΩ, and the resistance value and input value can be calculated:

$$R_t = \frac{1353*20400}{1353+20400} = 1268$$

For a polynomial function where a linear curve is considered, no complicated polynomial function is required.

The simple equation is

$$F(x) = a_1 * x + a_0, a_1 = 1, a_2 = 0, a_3 = 0, a_4 = 0, a_5 = 0$$

Defining the parameters a_0... a_5 will improve the accuracy by adding the sensor non-linearity in the temperature calculation.

Polynomial coefficients

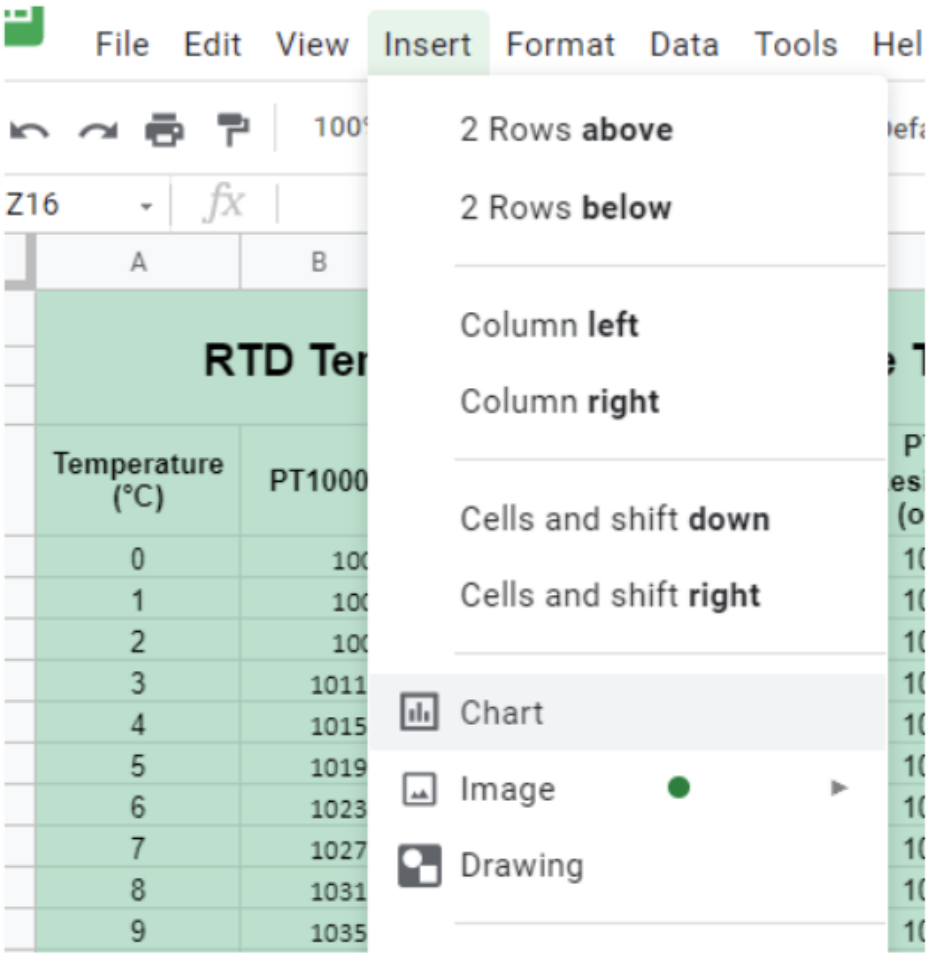
It would be best if you use a temperature, pressure, torque… Vs. Resistance tables for extracting the polynomial coefficients.

In this example, we are going to extract the polynomial coefficient.

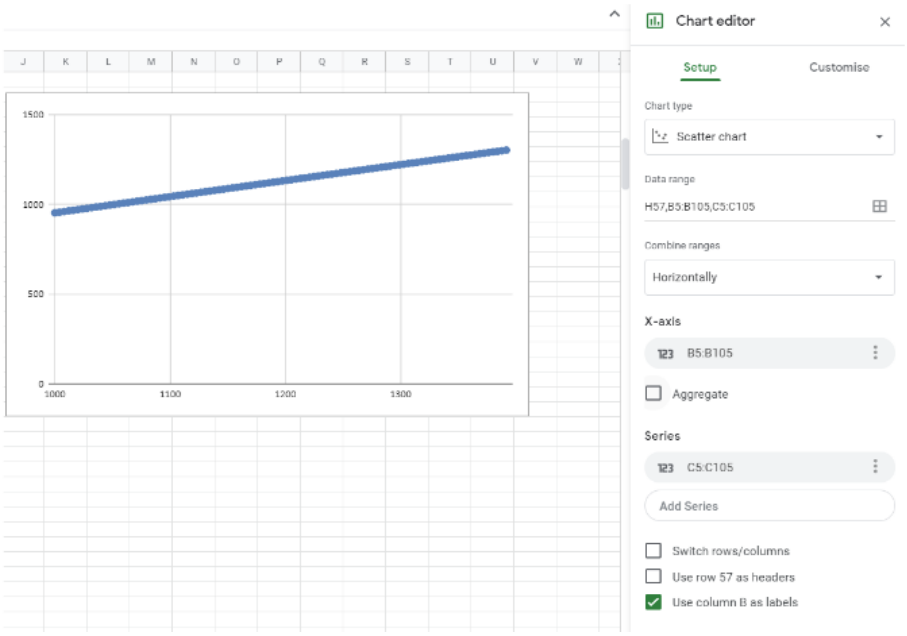
Follow the steps below for extracting the polynomial coefficient in Google spreadsheets:

1 - Create the table.

2 - Insert the chart.



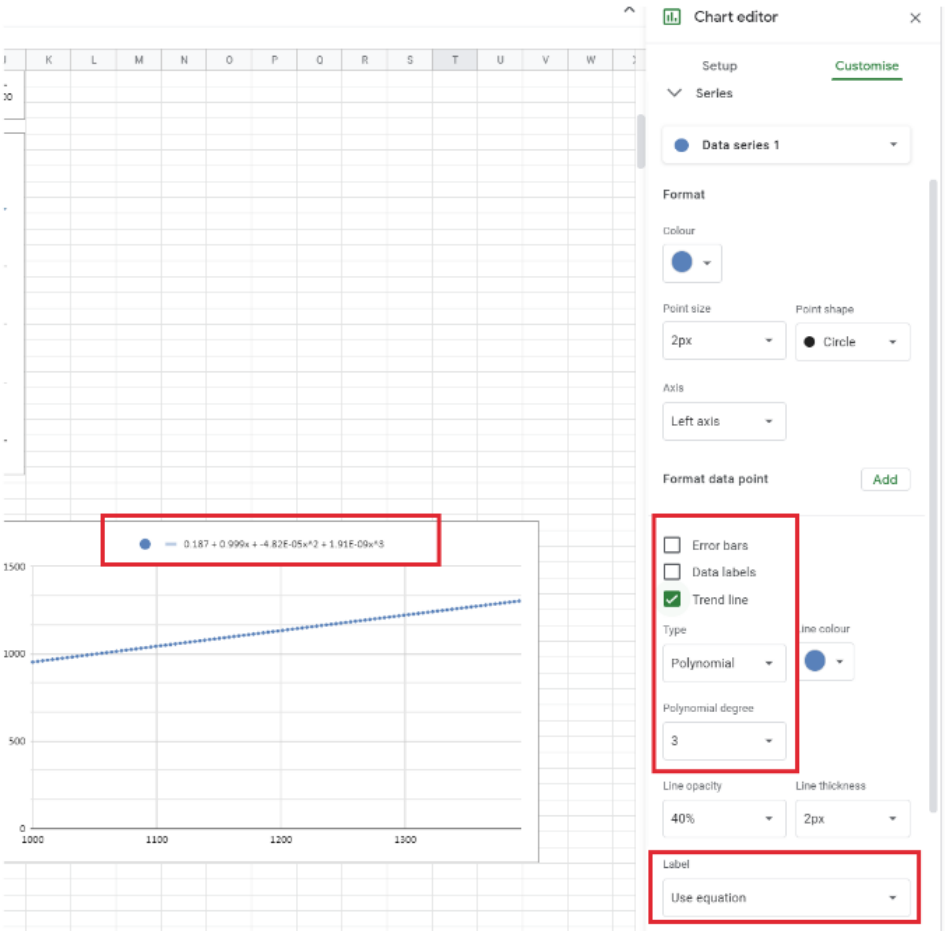
3 - Select the *Scatter chart* type in the chart editor menu in Setup.



4 - In the chart editor menu select the series then select *Trend lines*.

5 - Then you can define the type of the line (polynomial) and polynomial degree.

6 - Use the equation as the label, then the polynomial function will appear on top of it.



CANopen over EtherCAT CiA 402

Overview

SOMANET drives are based on and comply with different industry standards: EtherCAT, CANopen, CoE (CANopen over EtherCAT) and the CANopen device profile for motion drives.

EtherCAT is defined by the EtherCAT user group in the IEC-Standard 61158 and is defining the bus system and how the drive slave communicates with a master on the physical and basic communication level.

CANopen is standardized by CiA and the CiA 301 describes the CANopen application layer and the communication profile.

This describes mainly how to transfer content between a master and a slave and how to realize the real time requirements.

CoE is specifying how the CANopen protocol is used in the EtherCAT environment.

CiA 402 is the device profile for drives and motion control applications. CiA 402 defines how drives communicate with the masters:

The object number designates where to find or send the required information and all basic functionality, for example how to switch on a drive, how to recover faults, drive specific state machines.

Complying with CiA 402 makes drives interchangeable between different manufacturers because all drives work similarly and have the same basic data on the same address. Manufacturer specific areas define how to deal with special features.

Masters are based on these standards and offer an abstraction layer, so only minor knowledge about the standards is required. The system offers an interface (usually graphical), so that the system can be configured as needed. The user does not have to care about protocols, such as starting up the communication, fault and recovery procedures, PDO mapping, communication supervision, because it is already covered by the master. Usually, process values can be set by the user directly through the API of a master software without having to care about the details.

Nevertheless, having basic knowledge of the basic principles will help to establish the connection with these masters, bug fixing and fault diagnosis.

If no commercial master is used and the interface to a SOMANET drive is manually implemented or the system is realized with the use of free basic solutions like SOEM, a deep knowledge of the standard is required.

EtherCAT

EtherCAT (Ethernet for Control Automation Technology) is an Ethernet-based fieldbus system. The protocol is standardized in IEC 61158 and is suitable for both hard and soft real-time computing requirements in automation technology.

EtherCAT is defined by the ETG (EtherCAT Technology Group).

EtherCAT is able to deal with hard real time requirements, and the built-in synchronization mechanisms allow synchronized movement of multiple drives.

An EtherCAT network consists of an EtherCAT master and one or more EtherCAT slaves. SOMANET servo drive is an EtherCAT slave. To operate it, an EtherCAT master is required. Please refer to here for more details.

CANopen over EtherCAT (CoE)

CANopen is a communication protocol and device profile specification for embedded systems used in automation. The CANopen over EtherCAT protocol (CoE), which is used by all SOMANET servo drives, enables the complete CANopen profile family to be utilized via EtherCAT to ensure both high functionality and good communication performance.

CANopen is maintained by the CiA (CAN in Automation User Group) and is standardized by the European Norm EN 50325-4.

The CANopen protocol is not only restricted to physical CAN networks, but can as well be used on other field bus systems like EtherCAT or PowerLink.

Sufficient knowledge of CANopen concepts is also very important for correctly controlling our drives. The basic CANopen application layer and communication profile is specified in CiA 301. Based on this, the CANopen device profile for drives and motion control, which is known as CiA 402 whose 2.0 version is standardized as IEC 61800-7-201 from 2015, is released and conformed to by many servo drives including ours. It is recommended that users should read these standards from the official website of CAN in Automation before controlling SOMANET drives using a general EtherCAT master, while some essential concepts related to applications are introduced below as a quick guide.

CANopen basic Principles

The basic CANopen communication is defined in CiA 301.

All operable data in a CANopen device, including communication data and application data, is stored in an object dictionary. Each data can be accessed with a 16-bit index and an optional 8-bit subindex. For example, the measurement “Velocity actual value” is stored in 0x606C, and the con-

figuration “Brake output voltage” is stored in 0x2004:10. A complete object dictionary of SOMANET servo drives with firmware v5.0 can be found here.

Each CiA compliant device has a specific ESI file (EtherCAT Slave Information). Besides other device specific information, the ESI file contains the object dictionary.

To read/write data in a CANopen device, two methods as shown below are generally used.

SDO (Service Data Objects) can provide direct access to reading/writing any object entries in an object dictionary. However, this is done on demand and not cyclically, so it is mostly used for changing configurations, reading error codes or other actions on demand.

All objects can be read, but only some can be written as well (RO / read only <-> RW read write).

PDO (Process Data Objects) can provide real-time data transfer in a cyclic manner, so it is mainly used to transfer time-sensitive cyclical application data such as controlword/statusword, target/actual velocity etc. There are two kinds of PDOs: Transmit PDO (TPDO) and Receive PDO (RPDO) which provide slave-to-master and master-to-slave data transfer respectively. To configure a PDO, two objects in the object dictionary need to be configured by means of SDO. The first is the PDO parameter and the second PDO mapping. The PDO parameter describes the communication capabilities of the PDO while PDO mapping determines which object shall be mapped to this PDO to be transferred in real-time cyclically. By default, our standard PDO configuration already covers common application data so there is no need to configure from scratch.

CANopen Device Profile CiA 402 for drives and motion control

CANopen defines profiles for specific devices as IO-modules, sensors or drives. The profile for power drives is defined in CiA 402.

CiA 402 compliant drives have the same basic functionality and can easily be interchanged, even when they have manufacturer specific objects.

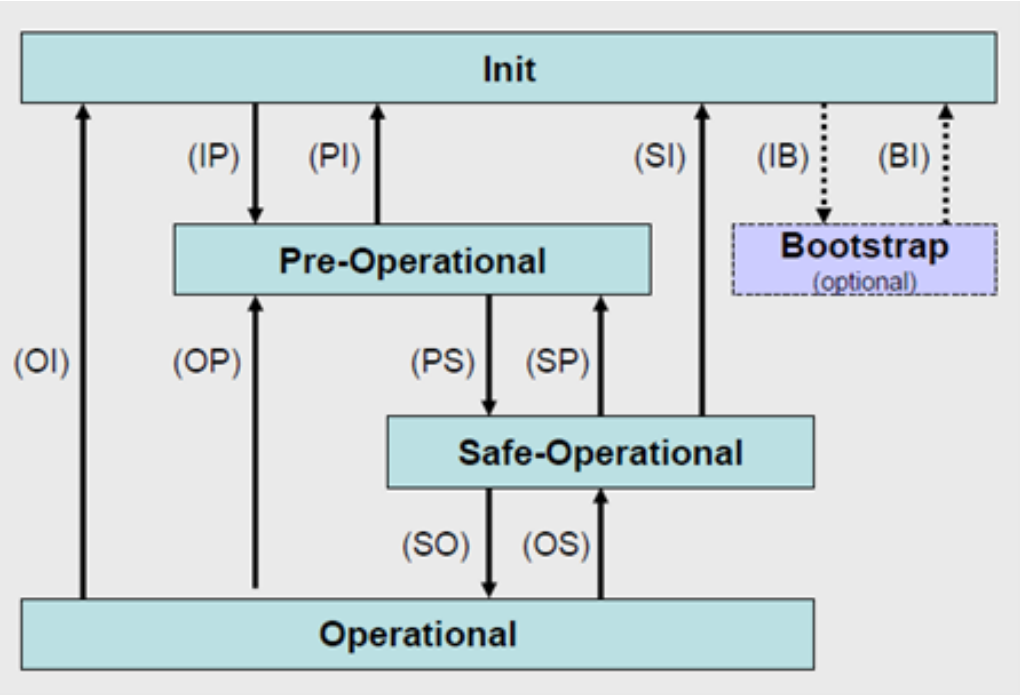
In CiA 402, it is specified that the behavior of a **Power Drive System (PDS)** should be controlled by a **Finite State Automaton (FSA)**. The PDS FSA has many states, and each state shall support a specific set of functions. For example, moving a motor is only possible in the state “Operation enabled”. State transitions can be triggered by writing the object “0x6040 Controlword” or internal fault signals (which transits the state to “Fault reaction active”) and current state of the FSA can be read from the object “0x6041 Statusword”. Please check our guide on status- and controlword for a detailed description and illustration.

The behavior of a Power Drive System is also determined by the activated mode of operation. The mode of operation can be selected by writing the object “0x6060 Modes of operation”. It defines how users would like to move the motor (for example, by setting the target

torque/velocity/position with a self-defined trajectory in profile modes, or by updating the torque/velocity/position cyclically in cyclic synchronous modes). The aforementioned Controlword and Statusword also have some specific operation mode-specific bits to extend functionalities. Please check this page for the definitions of all the modes of operation that SOMANET supports.

EtherCAT state machine

The figure below depicts the EtherCAT state machine and describes which communication is possible in which state.



The five states of the EtherCAT state machine and their transitions are shown below:

State	SDO (mailbox communication)	Slave-to-Master PDO (process data communication)	Master-to-Slave PDO (process data communication)
Init	No	No	No
Pre-Operational	Yes	No	No
Safe-Operational	Yes	Yes	No

State	SDO (mailbox communication)	Slave-to-Master PDO (process data communication)	Master-to-Slave PDO (process data communication)
Operational	Yes	Yes	Yes
Bootstrap	File access over EtherCAT (FoE) only	No	No

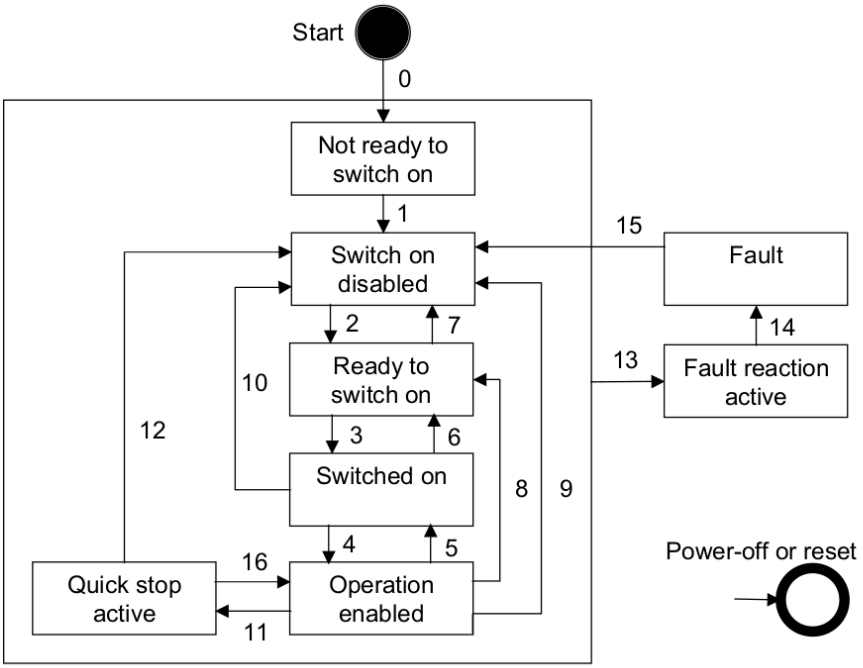
Note:

It is recommended to always change configurations in Pre-Operational state to avoid fault.

For more information, please refer to the official document of the EtherCAT state machine.

Drive state machine

The figure below specifies the drive state machine, with its state transitions based on user commands and internal drive faults.



Please refer to our guide on status- and controlword for details on how to manipulate the state machine.

PDO-Mapping

There is a standard set of PDO which consists of input objects (TxPDO) and output objects (RxPDO) in the firmware. The set covers almost all application data that will be used in real applications, generally no configuration is necessary.

However, if there are objects that are more suitable for the application, an individual set of PDO can be created by the following procedure:

1. Ensure that the desired object is PDO mappable (this can be checked in the object dictionary)
2. Switch EtherCAT state to Pre-operational
3. Disable PDO distribution by setting 0 to 0x1C12:00 (for RxPDO) or setting 0 to 0x1C13:00 (for TxPDO)
4. Disable one of the PDO mapping objects by setting its subindex 0 to 0
5. Enter the objects that should be mapped to the mapping entry subindices 1 to n
6. Set subindex 0 to the number n of used entries
7. Enable PDO distribution by setting 0x1C12:00 or 0x1C13:00 to their original values, by default 3 and 4 respectively

Changing the PDO mapping will not permanently be stored on the drive. The reconfiguration has to be done on every new start of the drive.

Many commercial masters provide the possibility to change the PDO mapping when configuring a device. The master will then change the PDO mapping automatically after every boot-up before enabling the operational state of the drive.

Communication Frequency and DC-Clock Synchronization

The communication frequency of EtherCAT systems is determined by the master's update rate, which is often represented as cycle time. Usually, the cycle time should be set between 250 µs and 8 ms. There are a few tips that may help to find the most appropriate cycle time based on the application and situation.

- When the cycle time is lower than 1 ms, a high performance of the hardware system is required.
- SOMANET drives have a sample time of 250 µs internally, so if the master's cycle time is set below this value, the application performance will generally not improve noticeably.

- In real-time applications, commands are sent to the slaves with a step size that depends on the master's update rate, so if the application requires fast command changes, a slow cycle time will have negative effects on control performance. If the control performance is limited by the hardware system, the control loop should be tuned relatively soft, or an FIR filter should be considered.

In EtherCAT systems, the clocks of master and slave(s) can be synchronized based on a Distributed Clock (DC) technology. This is especially important when multiple slaves are present and complex synchronous motions are required. When no distributed clock is available, the clocks on the slave and master devices will drift, and this will cause a cycle to periodically either miss a command, or complete before the command arrives at the controller. This will effectively either double the input command or see no change to the input command. Please refer to distributed clocks for more information.

Using Statusword and Controlword

Overview

What are Control- and Statusword?

Commands are sent by the master to the drive with the controlword and the drive sends information about its state back to the master with the statusword.

The controller is run through a state machine as defined in the CiA402 standard. The state changes are requested by the master using the controlword object 0x6040 and the slave will show the actual state in the statusword object 0x6041.

Only bits 0,1,2,3 and 7 in the controlword are used to change the state of the controller.

Only the first 7 bits of the statusword reflect the controller state.

How are they used?

The controlword and statusword should be used as defined in the CiA402 standard.

Manufacturer and Operation mode specific bits

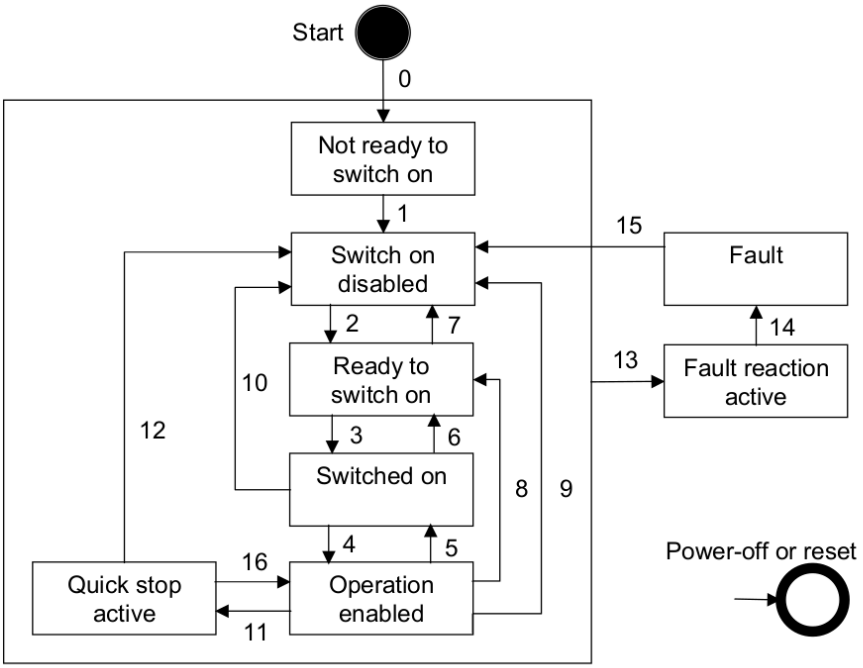
Some bits of controlword and statusword are dependent on the operation mode the drive is in, some bits of the statusword are also set by the manufacturer. Please check the section of each mode for details about which bits have which meaning:

- Cyclic synchronous torque control (CST)
- Cyclic synchronous velocity control (CSV)
- Cyclic synchronous position control (CSP)
- Profile position mode

- Profile velocity mode
- Profile torque mode
- Homing modes

Bits in Status- and Controlword

The finite-state automaton (FSA) shows different states of the drive and how the transitions between them can be executed.



Statusword

General structure

- The basic bits (0-7) are identical in each mode.
- The red marked bits 9 and 14-15 are not supported (N/A)
- The yellow marked bits have a specific functions which depends on the mode the drive is currently in. See the documentation of the specific mode for details.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ms		om-	om-	il-	tr	rm	ms	w	so-	qs	ve	f	oe	so	rt-

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		s	s	a					d						s-o
N/A		operation mode specific bits				N-/-A		basic							

The basic bits of the statusword indicate in which state the drive is:

State coding

Statusword	FSA state
xxxx xxxx x0xx 0000b	Not ready to switch on
xxxx xxxx x1xx 0000b	Switch on disabled
xxxx xxxx x01x 0001b	Ready to switch on
xxxx xxxx x01x 0011b	Switched on
xxxx xxxx x01x 0111b	Operation enabled
xxxx xxxx x00x 0111b	Quick stop active
xxxx xxxx x0xx 1111b	Fault reaction active
xxxx xxxx x0xx 1000b	Fault

Note:

- If bit 4 of the statusword is 1 that means that voltage is applied to the drive
- If bit 5 of the statusword is 0 that means that the drive is in quick stop active state
- If bit 7 of the statusword is 1 this indicates the presence of a warning condition
- If bit 11 is 1 this indicates that one of the control values has been limited. See table 1

Bits 9, 14 and 15 are not used

Legend

Abbr.	Meaning
rtso	ready to switch on
so	switched on
oe	operation enabled
f	fault
ve	voltage enabled
qs	quick stop
sod	switch on disabled
w	warning
ms	manufacturer-specific
rm	remote
tr	target reached
ila	internal limit active
oms	operation mode specific
ms	manufacturer specific

Mode specific bits of the Statusword

Mode of operation	13	12	11	10	8
Profile Position	Following error	Set-point acknowledge	Internal limit Active	Target reached	N/A
Profile Velocity	N/A	Speed bit			
Profile Torque		N/A			
Cyclic Synchronous Position		Target position ignored			
Cyclic Synchronous Velocity		Target velocity ignored			
Cyclic Synchronous Torque		Target Torque ignored			
Homing	Homing error	Homing attained			

Internal limit Active bit

The values that are limited by this bit differ in each mode:

Limiting object	Limited value	Operating modes
Max torque (0x6072)	Torque demand value (0x6074)	All
Max torque (0x6072)	• Target torque (0x6071) • Torque offset (0x60B2)	PT / CST
Max. motor speed (0x6080)	Velocity demand value (0x606B)	CSP / PP
Max. motor speed (0x6080)	Profile velocity (0x6081)	PP
Max. motor speed (0x6080)	Target velocity (0x60FF)Velocity offset (0x60B1).	CSV / PV
Software position limit (0x607D)	Position demand value (0x6062)	CSP/PP

Legend:

PP: Profile Position

PV: Profile Velocity

PT: Profile Torque

CSP: Cyclic Synchronous Position

CSV: Cyclic Synchronous Velocity

CST: Cyclic Synchronous Torque

Controlword

General structure

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ms	ms	ms	ms	ms	r	om-	h	fr	om-	om-	om-	eo	qs	ev	so

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						s			s	s	s				
N/A							basic		operation mode specific bits			basic			

> Note:
>
> bits 9,10,11,12,13,14 and 15 are not used.

Legend:

Abbr.	Meaning
so	switched on
ev	enable voltage
qs	quick stop
eo	enable operation
oms	operation mode specific
fr	fault reset
h	halt
oms	operation mode specific
r	reserved
ms	manufacturer specific

State Coding

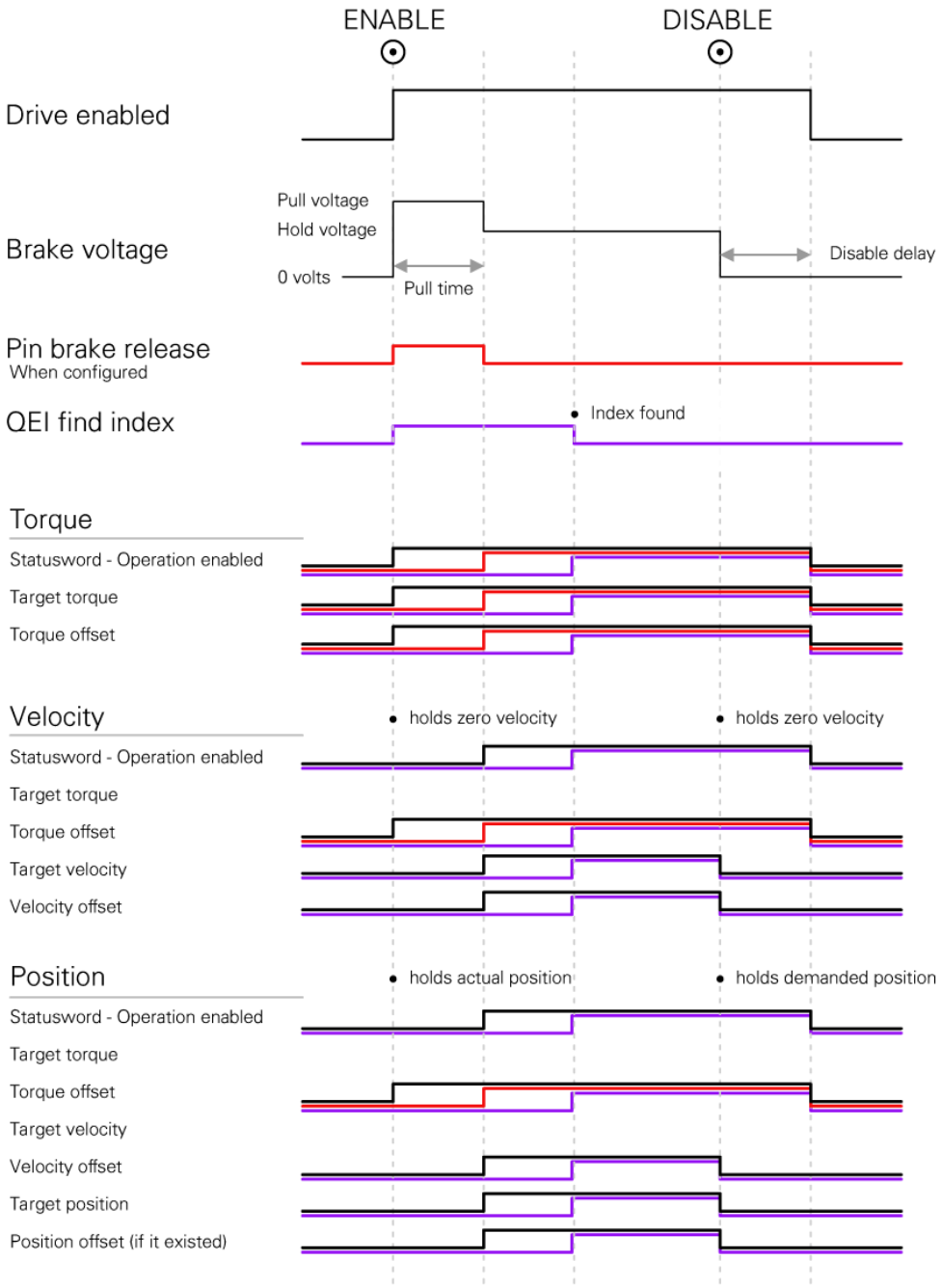
Com-mand	b7	b3	b2	b1	b0	Trans-itions
Shut-down	0	x	1	1	0	2,6,8
Switch on	0	0	1	1	1	3
Switch on + enable operation	0	1	1	1	1	3+4 (NOTE)
Disable voltage	0	x	x	0	x	7,9,10,12
quick stop	0	x	0	1	x	7,10,11
disable operation	0	0	1	1	1	5
enable operation	0	1	1	1	1	4,16
Fault reset	1	x	x	x	x	15
NOTE Automatic transition to Enable Operation state after executing SWITCHED ON state functionality.						

Mode specific bits of the Controlword

Mode of operation	8	7	6	5	4
Profile Pos-ition	Halt	N/A	N/A	Change set imme-	New set-point

Mode of operation	8	7	6	5	4
				diately	
Profile Velocity	Halt	N/A			
Profile Torque	Halt	N/A			
Cyclic Synchronous Position	No Operation mode specific bits used				
Cyclic Synchronous Velocity					
Cyclic Synchronous Torque					
Homing	Halt	N/A			Homing operation start

Timing Diagram



Replacing a SOMANET Drive

Overview

If a drive needs to be replaced, these steps are required to ensure the same system's behavior with the new drive.

In this document, the used configuration tool is OBLAC Drives. It can be used with an OBLAC Box or using a Virtual Machine.

Availability of the configuration file

All the parameters for a drive are stored in a config file. If a drive shows a fault and has to be replaced, this configuration can be exported and moved to the replacement drive. In some cases, the drive is completely broken and no connection is possible anymore. To prevent losing the configuration, it should be archived so it can be stored on a replacement.

The drive-specific commutation offset is also stored in the configuration file. Therefore, a specific config file is required **for each drive**.

If there are multiple similar motor drive setups with similar parameters, a common configuration file can be used. But in this case, the offset detection has to be done every time and can not be skipped.

> Attention:
>
> Operating the drive with a wrong commutation offset can lead to uncontrolled and dangerous movements of a motor!

If no backup configuration file is available, neither archived nor from the old drive, a completely new configuration with the configuration wizard has to be done.

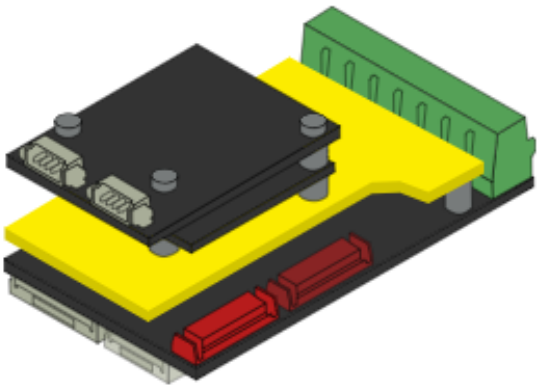
SOMANET Node with external Encoders

Save configuration file

Underneath the drive's picture on the main page of OBLAC Drives, different actions can be performed.

When clicking on "Save configuration file" a .csv file will be saved on the user's PC that contains different EtherCAT SDO objects with their actual values.

This file can be used to back up a specific configuration.





Replace the node

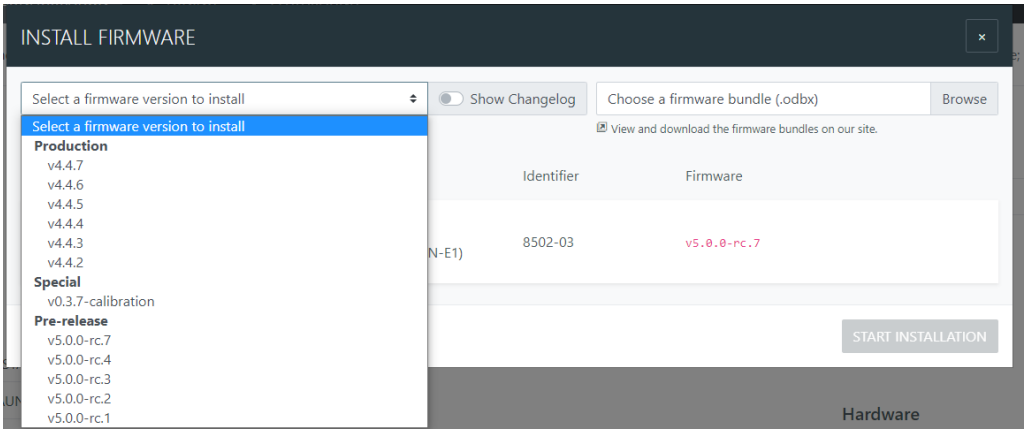
Before replacing the node, please note the firmware version that is currently installed.

When replacing the node, please make sure to have the encoders properly connected to the same encoder port.

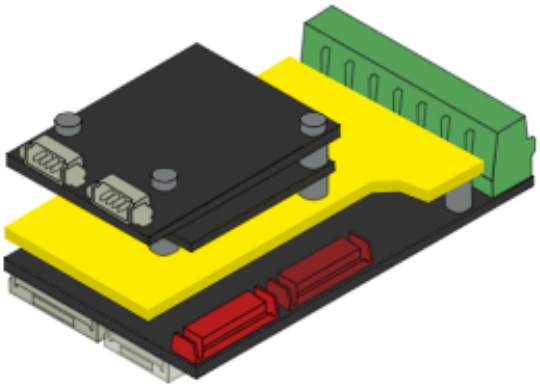
Update firmware

A firmware needs to be installed before the first commissioning of a new SOMANET drive. Please make sure to have the same or compatible firmware version installed as before.

The firmware update can be done online if the Virtual machine or the OBLAC Drives have internet connection, or it can be done offline through the firmware bundles that can be downloaded here



Load the configuration file





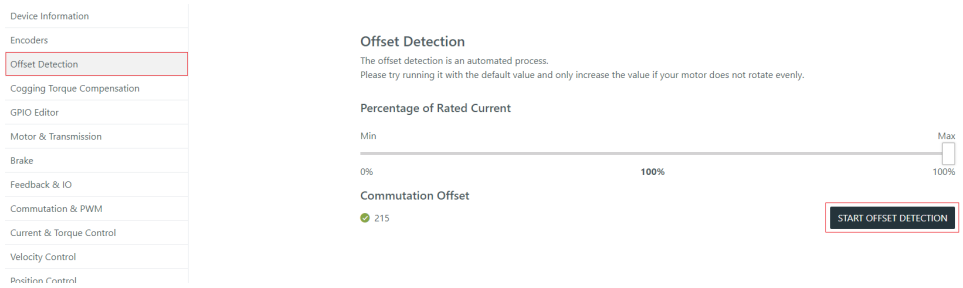
When clicking on the “LOAD CONFIGURATION FROM FILE” button, the following window will pop up. It contains all the objects that can be written from the .csv file to the drive’s memory. Please proceed by clicking on “LOAD CONFIGURATION” and wait until the procedure is finished.



Run offset detection

This step can be ignored, if the following points are true:

- the motor phase’s order is the same as the previous drive
- the motor encoder was not replaced
- the encoder mounting was not rotated.



Save the configuration to the drive

This step can be skipped if the previous step (Run offset detection) was skipped as well.

In case the offset detection was executed, please make sure to save the new parameters to the drive.

Saving configuration is done through the “SAVE TO DEVICE” green button in the top right corner of OBLAC Drives.



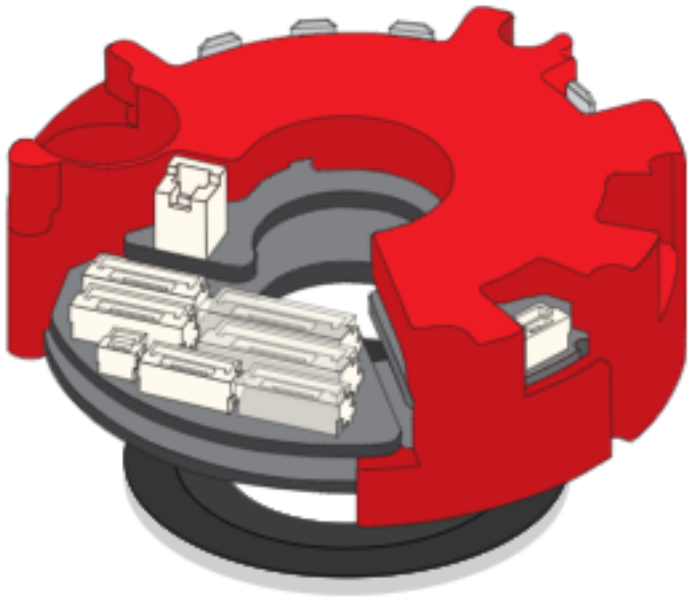
SOMANET Circulo with at least one internal encoder

Save the configuration file

Underneath the drive’s picture on the main page of OBLAC Drives, different actions can be performed.

When clicking on “Save configuration file” a .csv file will be saved on the user’s PC that contains different EtherCAT SDO objects with their actual values.

This file can be used to back up a specific configuration.



INSTALL FIRMWARE

LAUNCH SETUP WIZARD

LOAD CONFIGURATION FROM FILE (.csv)

SAVE CONFIGURATION FILE (.csv)

SAVE ETHERCAT SLAVE INFO. ESI (.xml)

SAVE PROBLEM REPORT (.zip)

VIEW DEVICE LOG

VIEW SYSTEM LOG

FACTORY RESET

Mark a Reference position on your axis

After replacing Circulo, there is a small chance that the zero position moved more than the mechanical mounting tolerance. To detect this, a reference position is required.

Replace Circulo

Before replacing Circulo, please note down the firmware version that is currently installed.

Make sure the encoders are properly connected to the same encoder ports before replacing the servo drive.

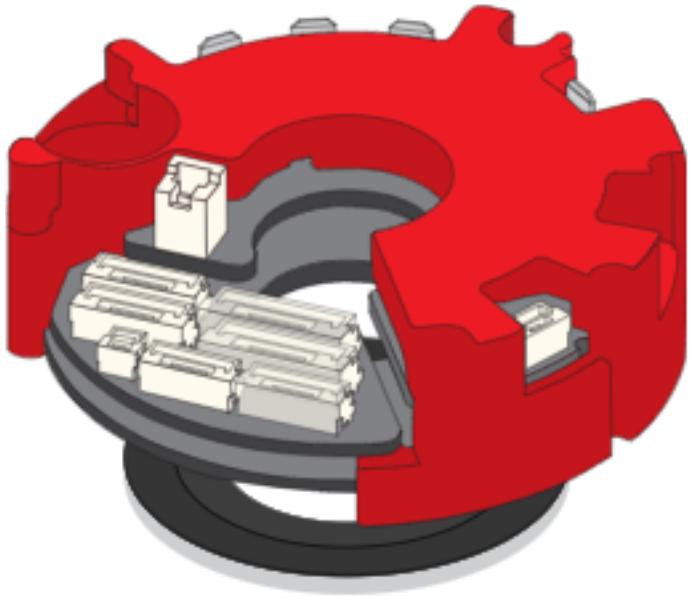
When using the Circulo internal encoder disks, please take the nominal encoder distances into account.

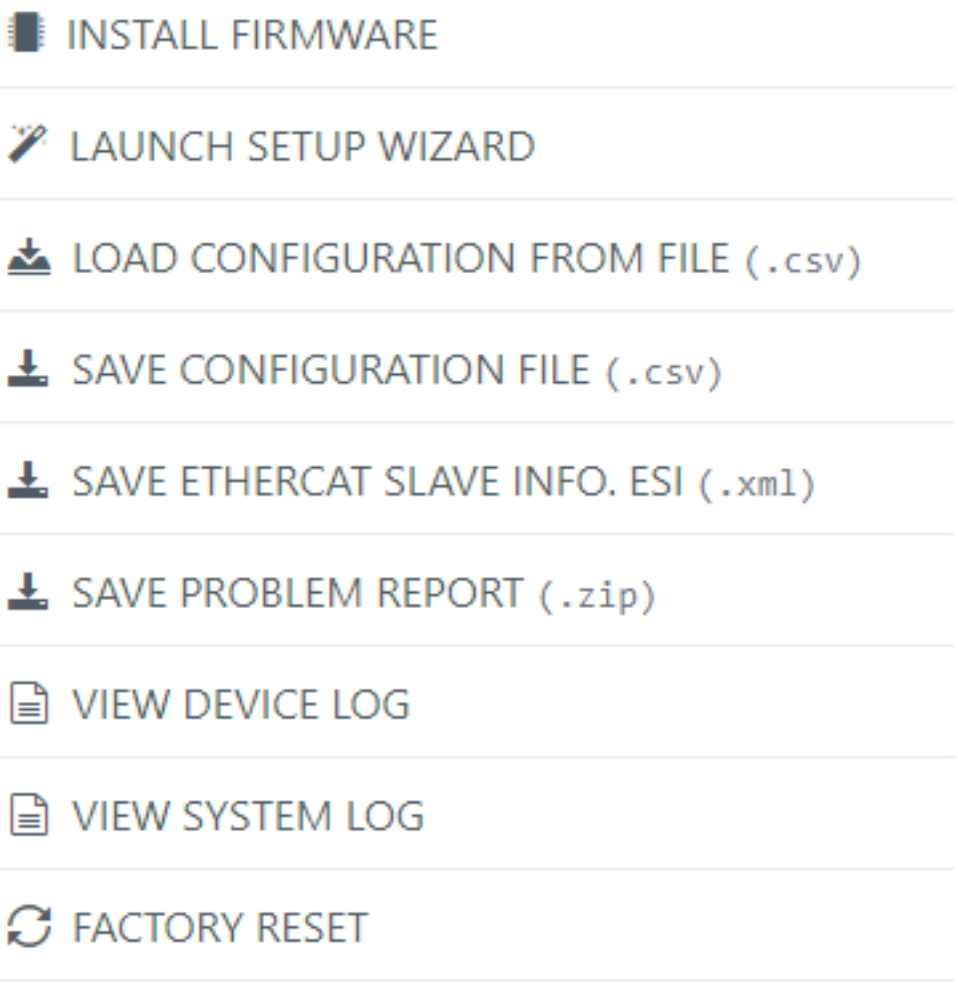
Load Calibration Firmware

The internal encoders have to be calibrated again to store the calibration data on the new drive. The calibration data is not stored in the config file nor somewhere else on the Circulo's file system and can't be moved from one to another Circulo.

> Note:
>
> Update to the latest calibration firmware.

Load the configuration file





Run offset detection

The commutation angle offset detection procedure has always to be executed.

Re-tune the drive

If the configuration of the drive has a sharply tuned velocity controller, a smoothly tuned velocity controller is required for the encoder calibration. If there is no smooth velocity while calibration or if the calibration fails, the velocity controller has to be temporarily re-tuned.

To re-tune the drive:

- Run system identification.
- Perform autotuning.
- Set the velocity controller bandwidth to a value lower than 30 Hz and test the velocity controller.
- Saving to the drive is not required, but the retuning will be lost after power-cycling.

> Attention:
> Tuning the drive will lead to motor movements. Place your system in a position where these movements can't damage anything. Please refer to the guidelines and restrictions in tuning in the motion control tuning part of our documentation.

Run calibration

- Execute encoder calibration
- check the phase error margin

When the calibration is finished, please update back to the production firmware.

> Attention:
> The calibration procedure will move the encoder for 2 full turns of the calibrated encoder! Place your system in a position where these movements can't damage anything. From OBLAC Drives version 21.3.0 onward, the movement will be limited within the defined software position limits.

Attention:

There is a small chance that the zero position of the encoder has changed after an encoder calibration. Therefore, an offset detection for the commutation and a re-referencing of the absolute encoder position are required after each calibration.

Load production firmware

Install the production firmware with the same or compatible version of the original drive.

Run offset detection and save offset to device

This step can be skipped if all the following points are considered:

- the original configuration of the replaced drive has been loaded to the new Circulo
- the encoder disks are not changed when Circulo was replaced
- the phase order of the motor is not changed

But running the offset detection after installing the production firmware will always be the safest way.

Save to device after offset detection



Test stable operation

Do some small movements in the playground. Start in torque mode to ensure a stable performance of the drive at the end of the replacement.

Check Zero Position

Verify that your zero did not move by checking the reference position. If the position moved or no reference position is available, you have to re-reference your system.

Tutorials

Tuning guides

Velocity PID Tuning

Position tuning with Gain Scheduling

Cascaded position Tuning

Drive operations through File Access over EtherCAT (FoE)

Tuning Guides

Those tuning guides shall give you a general impression of how to go through the tuning process of a SOMANET drive stack.

Click on one of the following links to get an overview of each tuning method.

Velocity PID Tuning

Position tuning with Gain Scheduling

Cascaded position Tuning

Velocity PID Tuning

In this section, the basic steps of tuning the velocity controller are explained.

PID Parameter Range

The constants k_P, k_I, and k_D should be between **0** and **10** with the precision of 6-digit floating point.

Measuring units of gains are: k_P [mNm/rpm],

k_I [mNm/(rpm*s]

k_D [mNm*s/rpm]

To find the proper values of PID constants of velocity controller the following steps are recommended:

Recommended Tuning Steps

Step 1.
: Check if all constants k_P, k_I, and k_D are set to **0**. After that activate the velocity controller. By default, the integral limit of velocity controller should be set to the maximum torque of selected motor in [milli-Nm].

Step 2.
: Set the reference velocity to 50% of rated motor velocity [rpm].

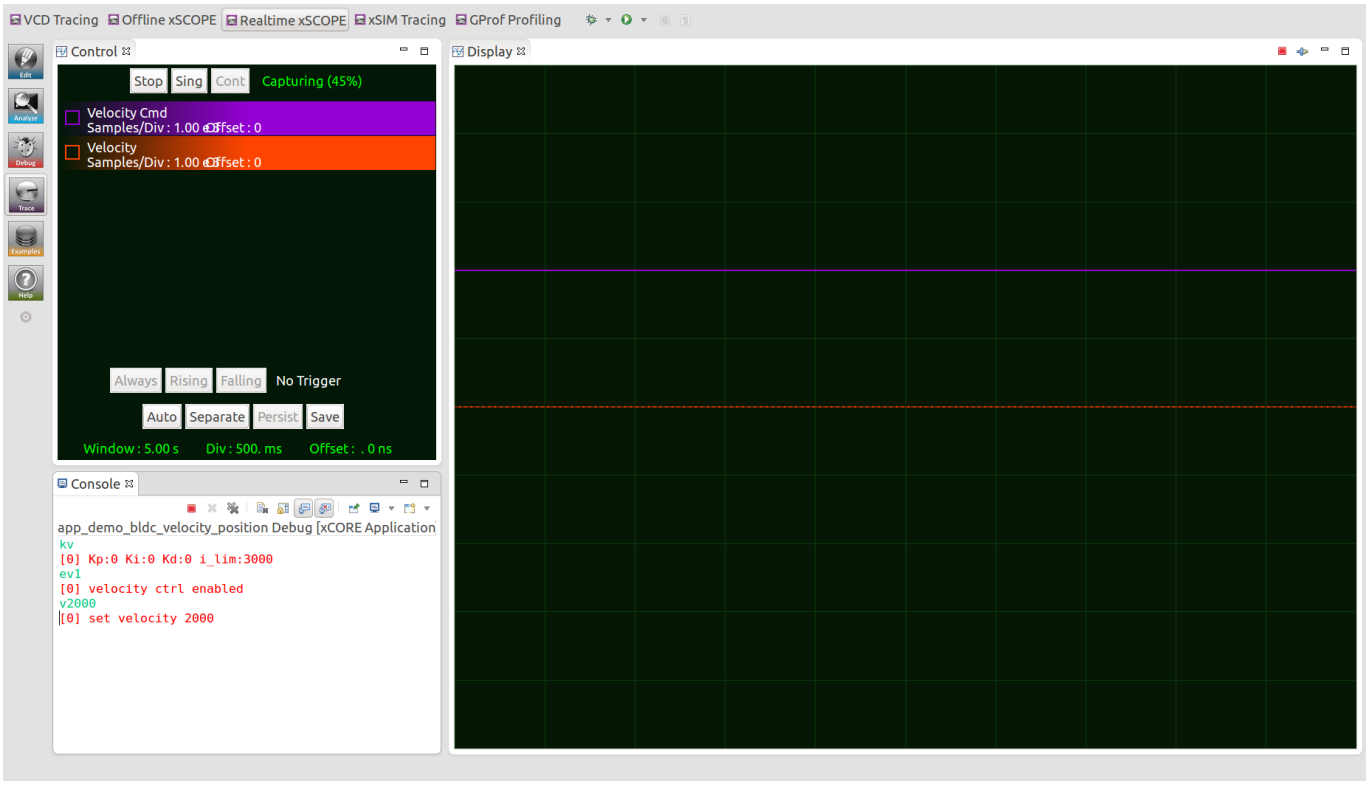


Fig. 1: setting the reference value of velocity to 50% of its rated value

Step 3.

Increase k_P until the measured speed is around 70% of the reference speed. As a suggestion start with k_P equal to 0.1, and in each step increase k_P with a factor of 2.

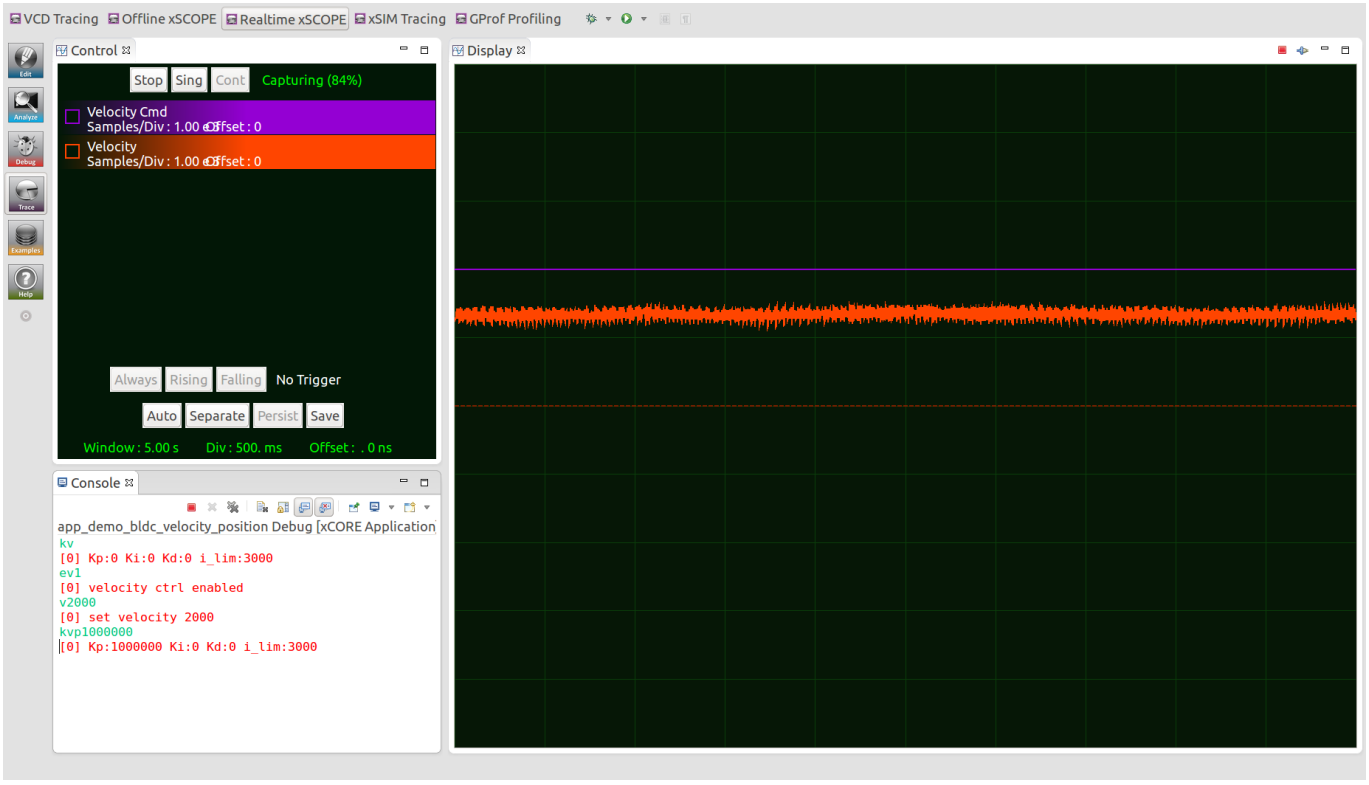


Fig. 2: increasing k_P to increase the actual speed

Step 4.

From this step forth, the step response of the system should be checked. Increase k_I step by step to increase the speed of the velocity controller (as a suggestion start with k_I equal to 0.01 and in each step increase it by a factor of 2).

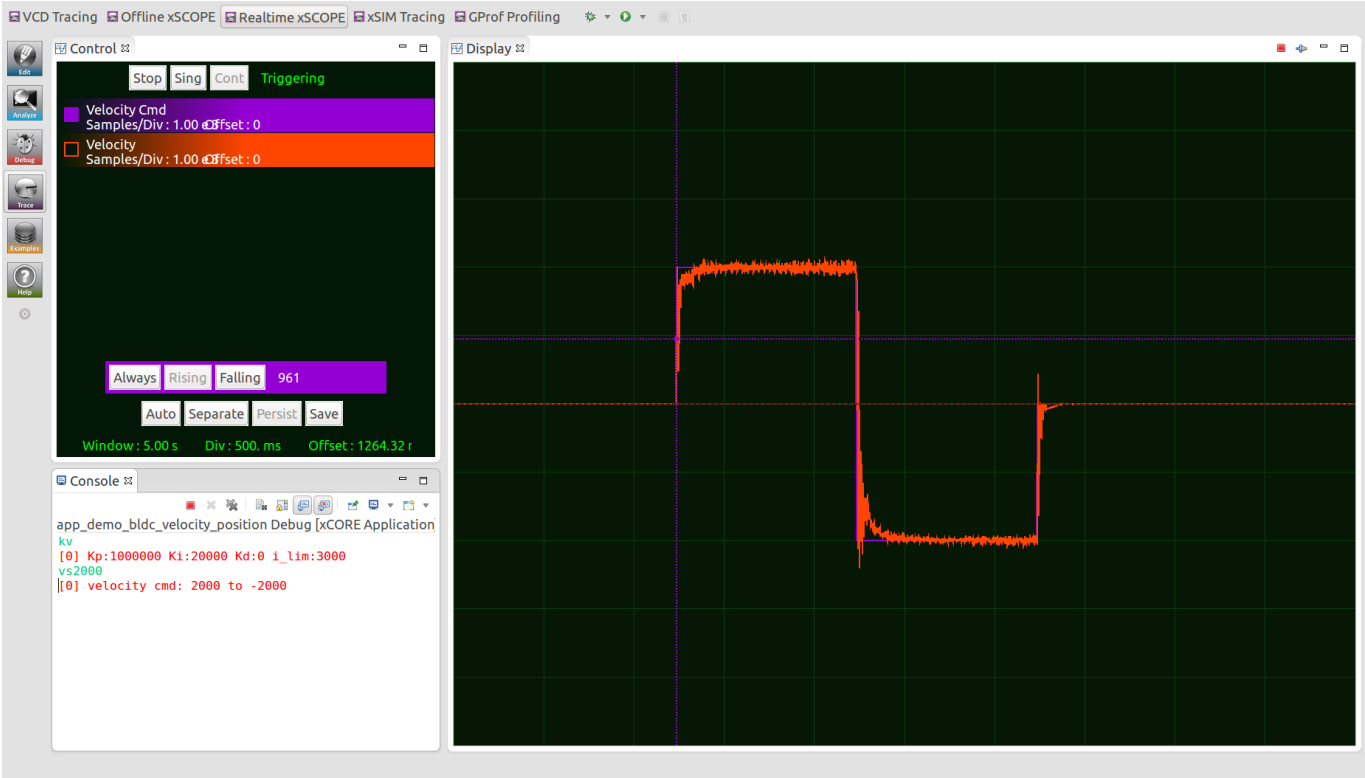


Fig. 3: Increasing k_I until a proper step response is generated by the velocity controller

At this point, the parameters k_P and k_I of the velocity controller are tuned. The parameter k_D of the velocity controller can be kept at its default value (0).

Parameter Tuning Guide for Position Controller with Gain Scheduling

Introduction

In this document, the steps of tuning the gain scheduling position controller are explained. The structure of the gain scheduling controller is the same as for a basic cascaded controller shown in Image 1.

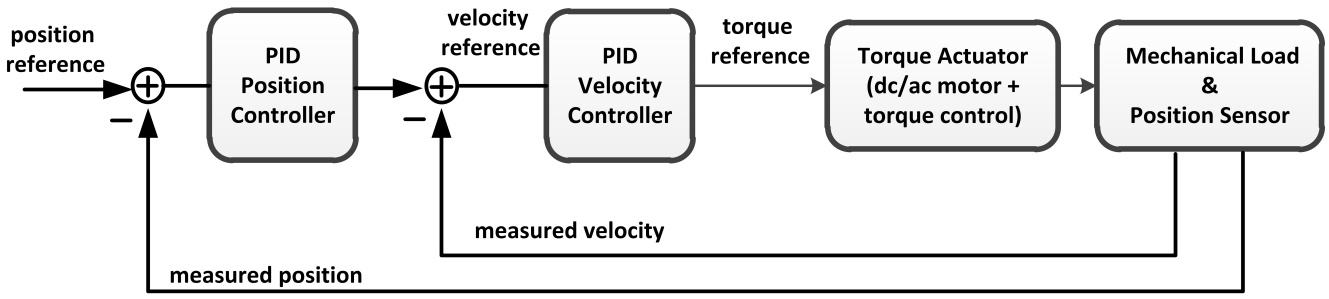


Image 1 - Cascaded position control structure

The gain scheduling controller changes gains based on the measured velocity of the axis. Basically it uses two sets of gains, one set is used when the system moves with low velocity and the other set is used for high velocities. User sets two parameters for low and high velocity area, i.e. low velocities are bounded with upper limit (low velocity limit) and high velocities are bounded with lower limit (high velocity limit). Between these limits , the gains are changed linearly. By setting the velocity limits, the user can control how often the gains will change, i.e. the user can adjust the behavior of the controller depending on the dynamics of the system.

The measuring units for the gains of the position controller are:

- k_P [rpm/ticks]
- k_I [rpm/ticks*s]
- k_D [rpm*s/ticks]

The measuring units for the gains of the velocity controller are:

- k_P [mNm/rpm]
- k_I [mNm/rpm*s]
- k_D [mNm*s/rpm]

Control Basics

Tuning of the gain scheduling controller is the same as for cascaded controller but it has to be done two times. One tuning should be done for the low velocity dynamics and the other one for high velocity dynamics. It is expected that the gains for lower velocities (axis is close to reference position) will be higher (controller reacts faster on overshoot and steady state error) then the gains for higher velocities (system is far away from the reference position). Usually the schedling of gains depending on axis velocity looks like on the picture shown in Image 2.

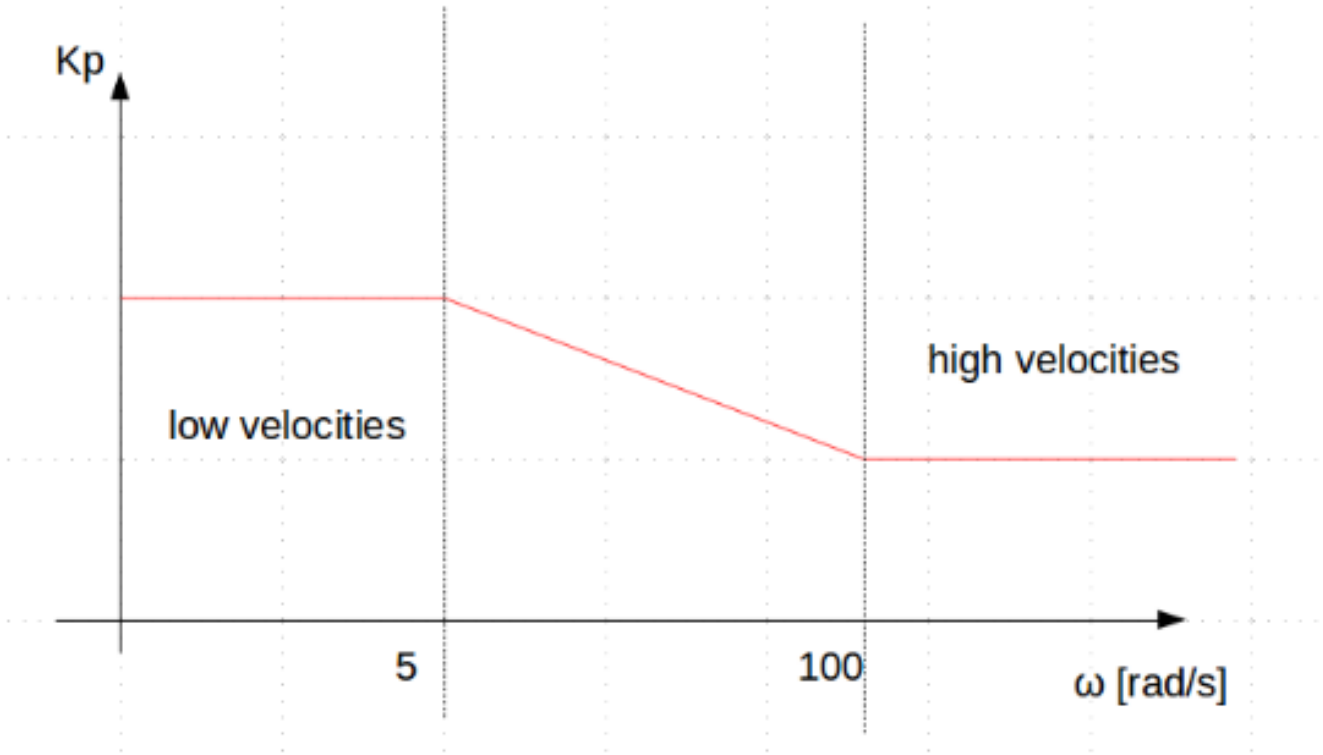


Image 2 - Scheduling of gains depending on velocity

Integral limit of the velocity controller should be set to the maximum torque of your torque actuator (in mNm) and integral limit of the position controller should be set to the maximum velocity of your system (in rpm).

Tuning Concept in Brief

Tuning steps decribed for cascaded controller should be applied for tuning the gain scheduling controller. First the controller should be tuned for high velocities and then for low ones. In the following section, the explained tuning concept is divided in separate systematic steps.

Tuning Steps

Step 1.

Set the reference step signal to a higher value, which will cause the axis to move with high velocity. Set the PID constants of both controllers equal to 0. By default, the integral limit of the velocity controller should be set to motor maximum torque in [mNm], and the integral limit of position controller should be set to motor maximum velocity in [rpm].

> Important:
>
> from this step forth, the step response of position controller should always be evaluated.

Step 2.

In this step, the k_P constant of the velocity controller should be tuned. Increase the k_P of both position and velocity controllers with a ratio of 10 ($k_{P_velocity}$ = 10 * $k_{P_position}$). Once the real position starts to follow the reference position, stop increasing the $k_{P_position}$ and only focus on increasing the $k_{P_velocity}$. Increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase $k_{P_velocity}$ until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is because of controller sharpness). Now, you can reduce $k_{P_velocity}$ to 90% of its value to increase the stability margin, and remove vibration noise.

Step 3.

Now that the velocity controller is tuned, it is time to tune the parameters of the PID position controller. Start with $k_{P_position}$, and increase it until you again get close to the instability margin, and then reduce $k_{P_position}$ to its 90%. So far, the proportional parts of both inner velocity loop and outer position loop are tuned.

Step 4.

In this step, increase k_I to eliminate the steady state error. As a suggestion start with k_I equal to 0.01, and in each step, increase it with a factor of 2. Increase k_I step by step until the following two conditions are met at the same time:

- the overshoot is in its acceptable range
- the steady-state error is eliminated in a short enough period of time

Step 5.

Repeat steps 1 - 4 for a small step reference signal (couple of thousands of ticks).

Step 6.

At this stage gains of the controller are determined. Beside gains user needs to determine the limit for low and high velocity dynamics. Recommendation is to observe the value of velocity during the tuning process. This should give a good basis for selection of those limits.

Parameter Tuning Guide for Position Controller with Cascaded Structure

Introduction

In this document, the basic steps of tuning PID position controller with cascaded structure are explained. As shown in Image 1, the cascaded structure is basically made of two controlling loops. The inner controlling loop is responsible for controlling the velocity while the outer controlling loop is responsible for controlling the position.

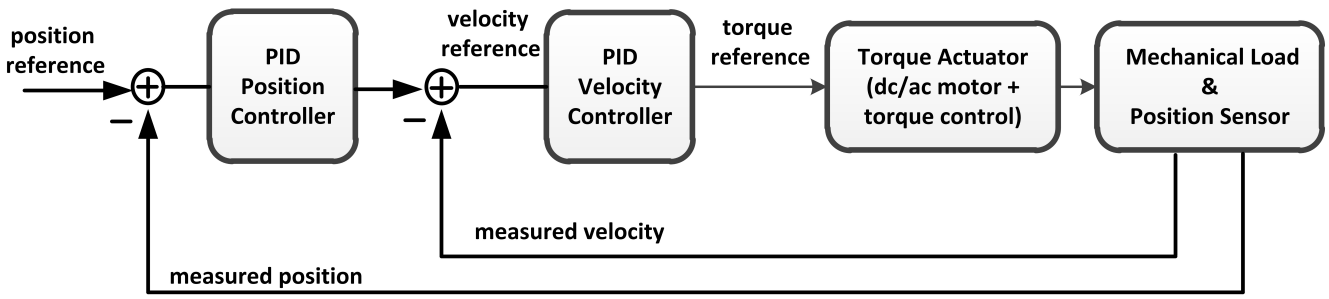


Image 1 - Cascaded position control structure

Measuring units of gains of position controller are:

- k_P [rpm/ticks]
- k_I [rpm/ticks*s]
- k_D [rpm*s/ticks]

Measuring units of gains of velocity controller are:

- k_P [mNm/rpm]
- k_I [mNm/rpm*s]
- k_D [mNm*s/rpm]

Control Basics

Tuning the position controller with cascaded structure is pretty straight forward. Cascaded controller is usually used in PI-P form, i.e. inner loop (velocity control loop) is only using the proportional part of its PID controller and the outer loop (position controlling loop) is only using its proportional and integral part. The other form is P-PI form, i.e. inner loop is using proportional and integral part and outer loop uses only proportional part. The inner velocity control loop is responsible for calculating the torque reference. As a result the integral limit of the velocity controller should be set to the maximum torque of your torque actuator (in mNm). Moreover, the outer position control loop is responsible for controlling the reference velocity, and consequently its integral limit should be set to the maximum velocity of your system (in rpm).

Tuning Concept in Brief

For tuning the cascaded structure, we should first focus on the inner (velocity) loop. As it is the inner controlling loop, it should be faster than the outer position loop. However, if all parameters of the outer position controller are set to 0, the inner loop will be disconnected from the user position commands. As a result, we can increase the k_P of both position and velocity controllers with a ratio of 10 ($k_{P_velocity}$ = 10 * $k_{P_position}$). Once the real position started to follow the reference

position, we can stop increasing the $k_{P_position}$, and only focus on increasing $k_{P_velocity}$. At this stage, we can increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase $k_{P_velocity}$ until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is caused by controller sharpness). Now, you can reduce $k_{P_velocity}$ to 90% of its value to increase the stability margin, and remove vibration noise.

Once the velocity controller (the inner controlling loop) is tuned, it is time to tune the position controller (the outer controlling loop). To tune the position controller loop, we should start with the P part of its PID controller. Increase $k_{P_position}$ until the entire position control gets close to instability margin. At this state, you will feel a vibration (or acoustic noise) which is because of too sharpened position control. At this step, reduce $k_{P_position}$ to its 90% to increase the stability margin and remove the vibration/ acoustic noise.

So far, the proportional parts of both inner velocity loop and outer position loop are tuned, and we can focus on integrator part of the outer position or inner velocity loop (depends on chosen structure of controller). Increasing the integrator constant will remove the steady state error, but at the same time, it adds some overshoot at step responses. As a rule of thumb, you can increase k_I step by step until the following two conditions are met at the same time:

- the steady-state error is eliminated in a short enough period of time
- the overshoot is in its acceptable range

In the following section, the explained tuning concept is divided in separate systematic steps.

Tuning Steps

Step 1.

Set the PID constants of both controllers equal to 0. By default, the integral limit of the velocity controller should be set to motor maximum torque in [mNm], and the integral limit of position controller should be set to motor maximum velocity in [rpm].

> Important:
>
> from this step forth, the step response of position controller should always be evaluated.

Step 2.

In this step, the k_P constant of the velocity controller should be tuned. Increase the k_P of both position and velocity controllers with a ratio of 10 ($k_{P_velocity} = 10 * k_{P_position}$). Once the real position starts to follow the reference position, stop increasing the $k_{P_position}$ and only focus on increasing the $k_{P_velocity}$. Increase (sharpen) the velocity controller as much

as possible. As a rule of thumb, increase $k_{P_velocity}$ until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is because of controller sharpness). Now, you can reduce $k_{P_velocity}$ to 90% of its value to increase the stability margin, and remove vibration noise.

Step 3.

Now that the velocity controller is tuned, it is time to tune the parameters of the PID position controller. Start with $k_{P_position}$, and increase it until you again get close to the instability margin, and then reduce $k_{P_position}$ to its 90%. So far, the proportional parts of both inner velocity loop and outer position loop are tuned.

Step 4.

In this step, increase k_I to eliminate the steady state error. As a suggestion start with k_I equal to 0.01, and in each step, increase it with a factor of 2. Increase k_I step by step until the following two conditions are met at the same time: - the overshoot is in its acceptable range - the steady-state error is eliminated in a short enough period of time

Drive operations through File Access over EtherCAT (FoE)

Overview

Synapticon SOMANET servo drives support File Access over EtherCAT (FoE) operations, which allows a number of operations for accessing files in the drive memory such as:

- Updating the firmware
- Uploading a drive configuration

> Note:
>
> The operation instructions are given based on a TwinCAT 3 environment. The steps are similar in other master applications.

Updating the firmware

> Attention:
>
> Installing firmware to a drive is a critical operation. Using a third-party tool (like a Master) can’t provide protection mechanisms such as version or hardware validation.

- Always use OBLAC Box for drive configurations unless it's unavoidable to use a third-party tool.
- Some critical items such as version compatibility need to be thoroughly checked to avoid unexpected results such as drive failure.

Prerequisite

If the drive is already used, updating the firmware with a third-party tool should be done only while the installed firmware and the firmware to install are checked by the Synapticon technical support team.

Factory resetting the drive before operations is recommended for safety reasons.

Setup Information

- The master environment used for the example is TwinCAT v3.1.4024.4
- The firmware used for the example is version v4.2.

The servo drive was factory reset before the operations.

Preparation

The appropriate firmware package needs to be provided by the support team.

Regarding the tool/master to be used, it should support File Access over EtherCAT (FoE) access and if necessary SII/EEPROM access.

Steps of Operation

> Attention:
>
> The following steps assume that the firmware has been checked for compatibility. The drive has been factory reset in this example.

> Note:
>
> Make sure to first contact the technical support team for thorough check and support.

1. Unzip the firmware package to the expected location. After the operation it should look like Image 1.

Name	Type	
app_motion-drive-v4.2.0-beta.12.bin	BIN File	Firmware binary
SOMANET_CiA_402.xml.zip	Compressed (zipped) Folder	EtherCAT Slave Information (ESI) file
somanet_cia402.sii	SII File	Slave Information Interface (SII) file
stack_image.svg.zip	Compressed (zipped) Folder	Stack image file

Image 1 - Files extracted from the firmware package

2. Switch the drive EtherCAT state to BOOT. In TwinCAT 3, the access (clicking “Bootstrap”) can be found as shown in Image 2 together with the FOE access.

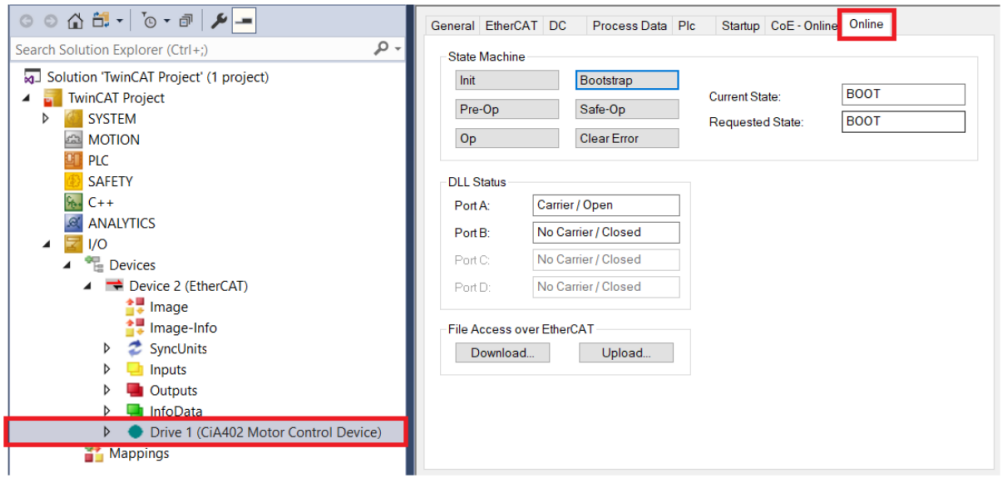


Image 2 - Accessing the slave operation in TwinCAT 3

3. In BOOT state, upload the firmware binary file to the drive memory through FOE, with the FOE name edited as “app_xyz.bin”. In TwinCAT 3 the operations are shown in Image 3-4.

This prompts the bootloader to flash the firmware.

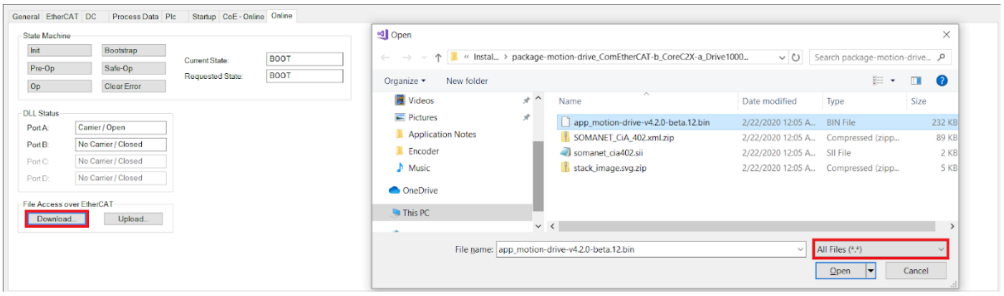


Image 3 - Choose the file to be uploaded to the slave

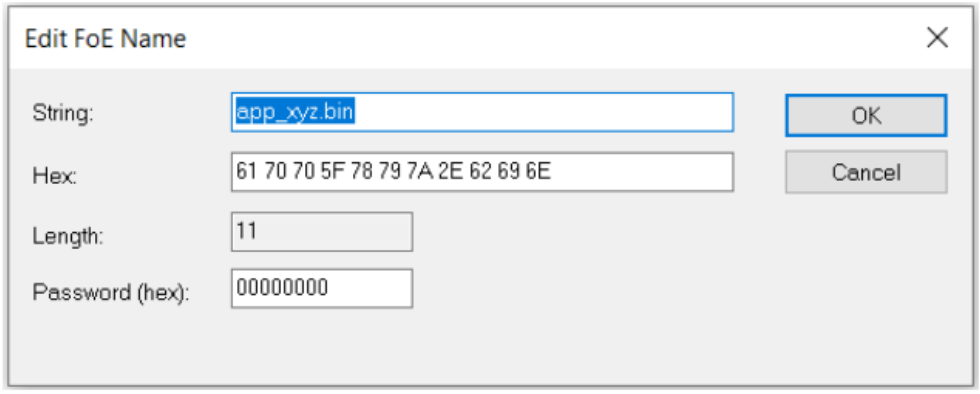


Image 4 - Edit FOE name

> Note:
> If OBLAC Box needs to be used for configuration, use the “INSTALL FIRMWARE” function and install the same firmware with OBLAC Box to ensure further compatibility with the tools.

Uploading a drive configuration

Setup Information

- The master environment used for the example is TwinCAT v3.1.4024.4
- The firmware used for the example is version v4.2.

> Note:
> Please make sure that the configuration file used is compatible with the firmware version installed.

Steps of Operation

1. In TwinCAT 3, double-click the servo drive under “I/O” - “Devices”, as shown in Image 5, to access the slave operations. Click the tab “Online” for state machine operations and FOE access.

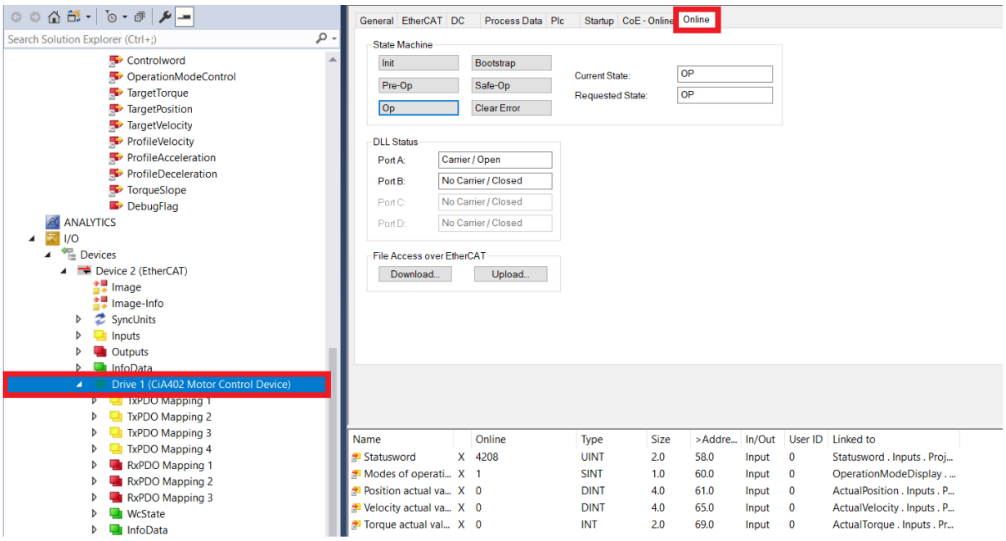


Image 5 - Accessing slave operation in TwinCAT 3

2. Switch the EtherCAT state of the drive to BOOT by clicking “Bootstrap”, this allows the following operations.

3. Click “Download” under “File Access over EtherCAT”, switch to browse “All Files”, and choose the configuration file to be uploaded, as shown in Image 6.

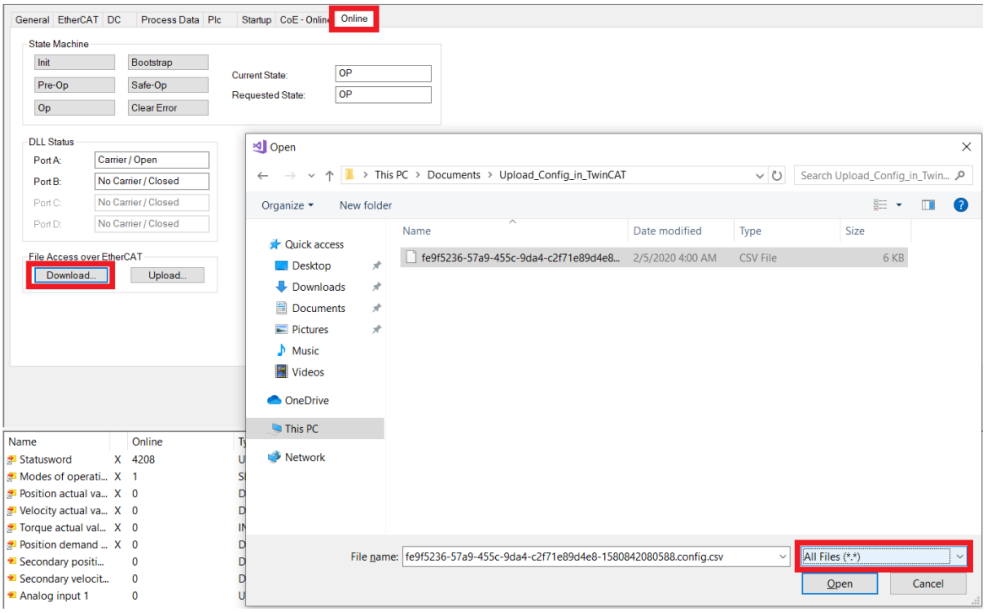


Image 6 - Choose the configuration file to be uploaded to the slave

4. Change the name to “config.csv” and click “OK” , as shown in Image 7.

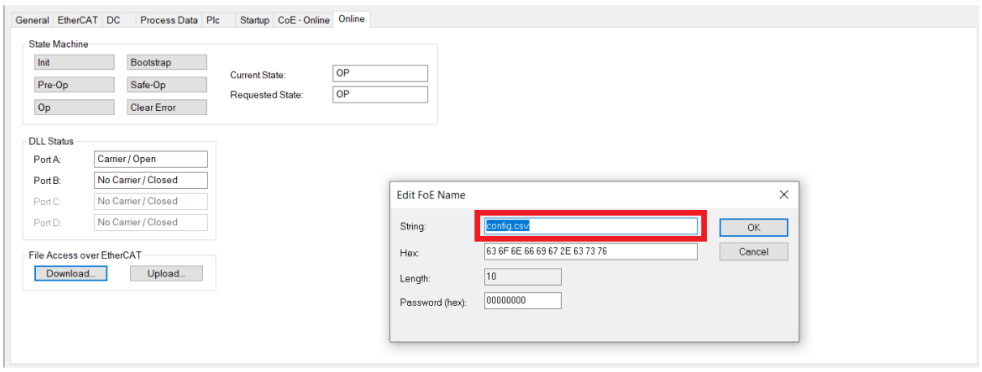


Image 7 - Edit FOE name

5. A progress bar can be seen at the bottom of the page in TwinCAT 3 after the FOE operation (see Image 8)



Image 8 - Progress bar of TwinCAT 3

6. The configuration file is successfully uploaded and stored in the permanent storage of the servo drive. It will be used once the drive is reinitialized at the transition from BOOT to INIT (or when the drive is power-cycled).

7. In case resetting or power-cycling the drive is not desired, object 0x1011 **("Restore default parameters")** allows loading from the permanent storage through CANopen over EtherCAT (CoE).

   Through SDO operation, read the current value of 0x1011:01 **("Restore all default parameters")**. Reading a value of "1" means the procedure is ready. Write value 0x64616f6c ("daol", "load" spelled backwards in ASCII) to 0x1011:01 to start the restore procedure.

   In TwinCAT 3, this can be done in the "CoE - Online" tab.

   Double-click on the object to write it, as shown in Image 9.

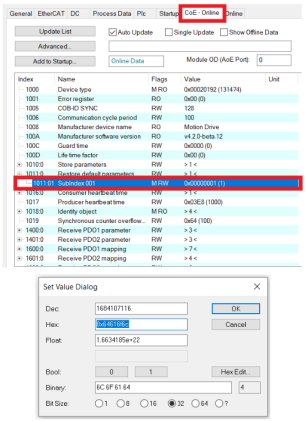
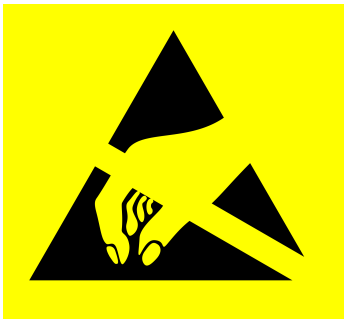


Image 9 - Restore default parameters without resetting drive

Handling instructions

Electrostatic discharge

All our products are sensitive electronic devices, please respect the following guidelines when handling them.



> Attention:
>
> Static electricity can damage electronic devices and your system. To avoid damage, keep static-sensitive devices in their static-protective bags until you are ready to install them.

To reduce the possibility of electrostatic discharge, observe the following precautions:

- Wear ESD protection gear such as wrist bands, ESD gloves, ESD shoes or use ESD-mats.
- Limit your movement. Movement can cause static electricity to build up around you.
- Handle the device carefully, holding it by its edges or frame.
- Do not touch solder joints, pins, or exposed printed circuitry.
- Do not leave the device where others can handle and possibly damage the device.
- While the device is still in its anti-static bag, touch it to an unpainted metal part of the system unit for at least 2 seconds. (This action removes static electricity from the package and from your body).
- Remove the device from its package and install it directly into your system, without putting it down. If it is necessary to put the device down, place it onto its static-protective bag. (If your device is an adapter, place it component-side up.) Do not place the device onto the cover of the system or onto a metal table.
- Take additional care when you handle devices during cold weather. Indoor humidity tends to decrease in cold weather, causing an increase in static electricity.

Magnetic sensitive devices

The encoder rings of “Sense” products are magnetized with a magnetic pattern that can be changed by various influences.



Special precautions must be taken to maintain the quality of the magnetic pattern permanently:

- Remove all strong magnets near the encoder rings - the stronger the magnet, the higher the risk of damage.
- Keep strong magnets away from the encoder rings even for short moments. The farther away, the lower is the effect on the rings.
- Make sure that no contact can occur between the encoder rings and strong permanent magnets or electromagnets during assembly.
- Avoid all materials between the encoder ring and, for example, a magnetic brake, being magnetic.
- When storing, supplying and picking parts, make sure the encoder rings are not transported or stored with other magnets. This includes permanent magnets for electric motors.
- Remove all tools with magnetic properties from the assembly area, including magnetized screwdrivers (often indicated by a black tip).
- Do not use magnet holders or handling magnets for the encoder rings.
- Remove all bags or garments with magnetic closures when working with the encoders rings.
- Avoid possible inductances caused by high voltage power lines near the encoder rings.
- Do not store encoder rings in a contiguous way as this may cause magnetic interference.
- Observe the minimal distance between the rings of 5 mm.

About this documentation

Original instructions

These are the original instructions of SOMANET Node .

Warnings

Throughout this documentation we use colored boxes that indicate potentially dangerous situations:

Note:
Background information that is not necessarily required for basic operation but for optimal performance.

Important:
Required for correct functionality. Ignoring these facts may result in erratic behavior or lack of function.

Attention:
Indicates a situation that could possibly lead to serious accidents and/or destruction of the servo drive.

Danger:
If these warnings are ignored, hazardous and potentially fatal situations may arise from improper or careless use.

Reporting errors

It is in our best interest to provide a detailed, user-centered and error-free documentation.

If anything is missing, not understandable or wrong, please use the red button at the bottom to send us feedback.

Versioning

This online documentation will be constantly updated and improved.

The version in the footer reflects the changes as follows:

Major.Minor.Patch

Patch: Corrections that only affect single sites. These can be changed values or other details.

Minor: New software features, products or sections.

Major: New feature of the documentation system.

> Note:
>
> Be careful when sharing links since the structure is constantly updated.